

Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος

Κάλλιπος Ανοιχτές Ακαδημαϊκές Εκδόσεις



Κριτικός Αναγνώστης: Δημήτριος Γκρίζαλης

Η φωτογραφία του εξώφυλλου έχει δημιουργηθεί χρησιμοποιώντας το DALL•E που βασίζεται στην αρχιτεκτονική GPT-4 της OpenAI.

ΘΕΜΕΛΙΩΣΕΙΣ ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΣΥΓΧΡΟΝΗΣ ΚΡΥΠΤΟΓΡΑΦΙΑΣ

Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας

Μια ανασκόπηση των κυριοτέρων εφαρμογών της σύγχρονης κρυπτογραφίας

Συγγραφική Ομάδα:

Γεώργιος Δροσάτος
Ινστιτούτο Επεξεργασίας του Λόγου
Ερευνητικό Κέντρο “Αθηνά”, Ξάνθη
Email: gdrosato@athenarc.gr

Ιωάννης Μαυρίδης
Τμήμα Εφαρμοσμένης Πληροφορικής
Πανεπιστήμιο Μακεδονίας, Θεσσαλονίκη
Email: mavridis@uom.edu.gr

Κωνσταντίνος Ράντος
Τμήμα Πληροφορικής, Σχολή Θετικών Επιστημών
Δημοκρίτειο Πανεπιστήμιο Θράκης, Καβάλα
Email: krantos@cs.duth.gr

Κριτικός Αναγνώστης:

Δημήτριος Γκρίτζαλης
Τμήμα Πληροφορικής
Οικονομικό Πανεπιστήμιο Αθηνών, Αθήνα
Email: dgrit@auerb.gr

Τίτλος πρωτότυπου: «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας»
Copyright © 2024, ΚΑΛΛΙΠΟΣ, ΑΝΟΙΚΤΕΣ ΑΚΑΔΗΜΑΪΚΕΣ ΕΚΔΟΣΕΙΣ



Το παρόν έργο διατίθεται με τους όρους της άδειας Creative Commons Αναφορά Δημιουργού – Μη Εμπορική Χρήση – Παρόμοια Διανομή 4.0. Για να δείτε τους όρους της άδειας αυτής επισκεφτείτε τον ιστότοπο <https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode.el>

Αν τυχόν κάποιο τμήμα του έργου διατίθεται με διαφορετικό καθεστώς αδειοδότησης, αυτό αναφέρεται ρητά και ειδικώς στην οικεία θέση.

Συντελεστές έκδοσης

Γλωσσική επιμέλεια:

Γ. Δροσάτος, Ι. Μαυρίδης, Κ. Ράντος

Γραφιστική επιμέλεια:

Γ. Δροσάτος, Ι. Μαυρίδης, Κ. Ράντος

Τεχνική επεξεργασία:

Γ. Δροσάτος, Ι. Μαυρίδης, Κ. Ράντος

ΚΑΛΛΙΠΟΣ

Εθνικό Μετσόβιο Πολυτεχνείο

Ηρώων Πολυτεχνείου 9

15780 Ζωγράφου

www.kallipos.gr

Βιβλιογραφική αναφορά:

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος, (2024). Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας. Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις

Διαθέσιμο στο:

<https://doi.org/10.xxxxx/978-618-85370-x-x>

ISBN:

978-618-85370-X-X

Αφιερώνεται
στους φοιτητές των Τμημάτων
Πληροφορικής και Ηλεκτρολόγων
Μηχανικών & Μηχανικών Υπολογιστών
της Ελλάδος

ΠΕΡΙΕΧΟΜΕΝΑ

Κατάλογος Σχημάτων	xi
Κατάλογος Πινάκων	xv
Πρόλογος	xvii

I Βασικές Θεμελιώσεις και Τεχνικές	1
1 Κρυπτογραφία Συμμετρικού Κλειδιού	3
1.1 Εισαγωγή	3
1.2 Βασικές Έννοιες	4
1.2.1 Αντικατάσταση	4
1.2.2 Αντιμετάθεση	5
1.3 Κρυπτογραφικοί Αλγόριθμοι Μπλοκ	5
1.3.1 Κώδικας Feistel	7
1.3.2 AES	8
1.3.3 Camellia	11
1.3.4 Serpent	11
1.4 Τρόποι Λειτουργίας Αλγορίθμων Μπλοκ	11
1.4.1 ECB	11
1.4.2 CBC	13
1.4.3 CTR	15
1.4.4 CFB	16
1.4.5 OFB	16
1.5 Αλγόριθμοι Κρυπτογράφησης Poής	18
1.5.1 HC-128	19
1.5.2 Salsa20/20 και ChaCha	21
1.5.3 SNOW 2.0	22
1.5.4 SNOW 3G	22

1.5.5	SOSEMANUK	22
1.5.6	Αλγόριθμοι Ροής Παλαιού Τύπου	23
1.5.6.1	Grain, Grain-v1 και Grain-128a	23
1.5.6.2	Mickey 2.0	23
1.5.6.3	Rabbit	23
1.5.6.4	Trivium	23
1.5.6.5	Μη ασφαλείς αλγόριθμοι κρυπτογράφησης ροής	24
1.6	Ασκήσεις-Εργασίες	25
	Εργασίες	25
2	Κρυπτογραφία Δημοσίου Κλειδιού	29
2.1	Ορισμός της Κρυπτογραφίας Δημοσίου Κλειδιού	29
2.2	Κρυπτοσύστημα RSA	30
2.2.1	Δημιουργία Κλειδιών	31
2.2.2	Αλγόριθμος Κρυπτογράφησης	31
2.2.3	Αλγόριθμος Αποκρυπτογράφησης	31
2.2.4	Δημιουργία και Επαλήθευση Υπογραφών	32
2.2.5	Ασφάλεια του RSA	32
2.3	Κρυπτοσύστημα ElGamal	34
2.3.1	Δημιουργία Κλειδιών	34
2.3.2	Αλγόριθμος Κρυπτογράφησης	34
2.3.3	Αλγόριθμος Αποκρυπτογράφησης	35
2.3.4	Δημιουργία και Επαλήθευση Υπογραφών	35
2.3.5	Ασφάλεια του ElGamal	36
2.4	Κρυπτοσύστημα Paillier	36
2.4.1	Δημιουργία Κλειδιών	36
2.4.2	Αλγόριθμος Κρυπτογράφησης	37
2.4.3	Αλγόριθμος Αποκρυπτογράφησης	37
2.4.4	Ασφάλεια του Paillier	37
2.5	Αλγόριθμος Ψηφιακών Υπογραφών DSA	38
2.5.1	Δημιουργία Κλειδιών	38
2.5.2	Δημιουργία και Επαλήθευση Υπογραφών	39
2.6	Κρυπτογραφία Ελλειπτικών Καμπυλών	39
2.6.1	Υπόβαθρο Ελλειπτικών Καμπυλών	40
2.6.1.1	Τύποι Ελλειπτικών Καμπυλών	40
2.6.1.2	Παράμετροι Τομέα	41
2.6.2	Κρυπτογραφικό Σχήμα ECIES	42
2.6.2.1	Προαπαιτούμενα Κρυπτογράφησης	42
2.6.2.2	Αλγόριθμος Κρυπτογράφησης	43
2.6.2.3	Αλγόριθμος Αποκρυπτογράφησης	43
2.6.3	Αλγόριθμος Ψηφιακών Υπογραφών ECDSA	43
2.6.3.1	Προαπαιτούμενα Δημιουργίας Υπογραφής	44
2.6.3.2	Δημιουργία Υπογραφής	44
2.6.3.3	Επαλήθευση Υπογραφής	44
2.7	Ασκήσεις - Εργασίες	45
	Ασκήσεις	45
	Εργασίες	46
3	Συναρτήσεις Σύνοψης	51

3.1	Ορισμός των Συναρτήσεων Σύνοψης	51
3.2	Δομικά Στοιχεία των Συναρτήσεων Σύνοψης	53
3.2.1	Κατασκευή Merkle-Damgård	53
3.2.1.1	Πλήρωση Δεδομένων	53
3.2.1.2	Συμπίεση Δεδομένων	53
3.2.2	Κατασκευή Wide-Pipe	54
3.2.3	Κατασκευή HAIFA	54
3.2.4	Κατασκευή Sponge	55
3.2.5	Συναρτήσεις Συμπίεσης με Χρήση Κρυπτογράφησης Μπλοκ	56
3.2.6	Δένδρα Merkle	57
3.3	Συνάρτηση Σύνοψης MD5	59
3.4	Συνάρτηση Σύνοψης Whirlpool	60
3.5	Οικογένεια Συναρτήσεων Σύνοψης SHA	61
3.5.1	Συνάρτηση Σύνοψης SHA-1	62
3.5.2	Συνάρτηση Σύνοψης SHA-2	63
3.5.3	Συνάρτηση Σύνοψης SHA-3	65
3.6	Οικογένεια Συναρτήσεων Σύνοψης BLAKE	66
3.6.1	Συνάρτηση Σύνοψης BLAKE	67
3.6.2	Συνάρτηση Σύνοψης BLAKE2	69
3.6.3	Συνάρτηση Σύνοψης BLAKE3	70
3.7	Ασκήσεις-Εργασίες	70
	Ασκήσεις	70
	Εργασίες	71
II	Μηχανισμοί και Πρωτόκολλα	75
4	Γεννήτριες (Ψευδο)Τυχαίων Αριθμών	77
4.1	Τυχαίοι Αριθμοί	78
4.2	Ορολογία	78
4.3	Αρχιτεκτονική των PRNG	79
4.4	Απαίτησεις Ασφαλείας για PRNG	81
4.5	Θέματα Γλοποίησης	82
4.5.1	Πηγές Εντροπίας	82
4.5.2	Εκτίμηση Εντροπίας	82
4.5.3	Αρχικοποίηση Γεννήτριας	82
4.5.4	Κρυπτογραφικά Ασφαλείς Γεννήτριες	83
4.6	Μηχανισμοί Κρυπτογραφικά Ασφαλών PRNG	83
4.6.1	Μηχανισμός HASH-DRBG	84
4.6.2	Μηχανισμός HMAC-DRBG	85
4.6.3	Μηχανισμός CTR-DRBG	87
4.7	Άλλοι Αλγόριθμοι PRNG	88
4.7.1	Αλγόριθμος Blum-Blum-Shub	89
4.7.2	Αλγόριθμος Blum-Micali	91
4.7.3	Αλγόριθμος Yarrow	91
4.8	Ασκήσεις-Εργασίες	92
	Ασκήσεις	92
	Εργασίες	93

5 Ακεραιότητα και Αυθεντικοποίηση Δεδομένων	97
5.1 Εισαγωγή	97
5.2 Κώδικας Αυθεντικοποίησης Μηνύματος	98
5.2.1 MAC Βασισμένα σε Κρυπτογραφικούς Αλγορίθμους Μπλοκ	99
5.2.1.1 EMAC	100
5.2.1.2 AMAC	101
5.2.1.3 CMAC	101
5.2.2 MAC Βασισμένα σε Συναρτήσεις Σύνοψης	103
5.2.2.1 Επίθεση Επέκτασης Μήκους	104
5.2.3 Μέγεθος MAC	105
5.3 Αυθεντικοποιημένη Κρυπτογράφηση	105
5.3.1 Μηχανισμοί Γενικής Χρήσης	105
5.3.2 OCB	106
5.3.3 CCM	108
5.3.4 EAX	108
5.3.5 CWC	108
5.3.6 GCM	108
5.3.6.1 Συνάρτηση GHASH	110
5.3.6.2 Συνάρτηση GCTR	112
5.3.7 ChaCha20+Poly1305	112
5.4 Ψηφιακές Υπογραφές	112
5.4.1 RSA-PKCS #1	114
5.4.2 RSA-PSS	115
5.4.2.1 Mask Generation Function	117
5.4.2.2 RSASP1	117
5.4.3 (EC)DSA	118
5.4.4 PV Signatures	118
5.5 Ασκήσεις-Εργασίες	119
Εργασίες	119
6 Διαχείριση Κρυπτογραφικών Κλειδιών	125
6.1 Διαχείριση Κλειδιών	125
6.1.1 Φάση Προ-Λειτουργίας	126
6.1.2 Φάση Λειτουργίας	127
6.1.3 Φάση Μετα-Λειτουργίας	127
6.1.4 Φάση Καταστροφής	128
6.2 Απαίτησεις Προστασίας Κρυπτογραφικών Κλειδιών	128
6.3 Κρυπτοπερίοδος	129
6.3.1 Παράμετροι που Επηρεάζουν τη Διάρκεια	129
6.3.2 Κρυπτοπερίοδοι Συμμετρικών Κλειδιών	130
6.3.3 Κρυπτοπερίοδοι Ασύμμετρων Κλειδιών	131
6.4 Διαχείριση Συμμετρικών Κλειδιών	131
6.4.1 Δημιουργία και Διαμοιρασμό Συμμετρικών Κλειδιών	131
6.4.1.1 Δημιουργία Συμμετρικών Κλειδιών	131
6.4.1.2 Διαμοιρασμός Κλειδιών	132
6.5 Πρωτόκολλα Εδραίωσης Κλειδιών	132
6.5.1 Κλειδιά Συνεδρίας	133
6.5.2 Πρωτόκολλα Συμφωνίας Κλειδιού	133
6.5.2.1 Πρωτόκολλο Diffie-Hellman	134

6.5.2.2	Kerberos	135
6.5.3	Πρωτόκολλα Μεταφοράς Κλειδιών	138
6.5.3.1	Πρωτόκολλο Needham-Schroeder	139
6.6	Σχήματα Κοινής Χρήσης Μυστικών	140
6.6.1	Σχήμα Κοινής Χρήσης Μυστικών Ito-Nishizeki-Saito	142
6.6.2	Επαναλαμβανόμενη Κοινή Χρήση Μυστικών	143
6.7	Μονάδες Ασφαλείας Υλισμικού	144
6.8	Συστήματα Διαχείρισης Κλειδιών	145
7	Υποδομές Δημοσίου Κλειδιού και Υπηρεσίες Εμπιστοσύνης	149
7.1	Υποδομές Δημοσίου Κλειδιού	149
7.1.1	Μοντέλα ΥΔΚ	150
7.1.2	Ψηφιακά Πιστοποιητικά	152
7.1.2.1	Αναγκαιότητα Χρήσης Ψηφιακών Πιστοποιητικών	152
7.1.2.2	Πιστοποιητικά X.509 v3	153
7.1.3	Έλεγχος Κατάστασης Πιστοποιητικών	155
7.1.3.1	Λίστες Ανακληθέντων Πιστοποιητικών	155
7.1.3.2	Ηλεκτρονικό Πρωτόκολλο Κατάστασης Πιστοποιητικού	156
7.1.4	Υπηρεσίες Ανάκτησης Κλειδιών	156
7.1.5	Τύποι και Μεγέθη Κλειδιών	157
7.2	eIDAS	157
7.2.1	Τύποι Ηλεκτρονικών Υπογραφών κατά eIDAS	159
7.2.1.1	Ηλεκτρονικές Υπογραφές	159
7.2.1.2	Προηγμένες Ηλεκτρονικές Υπογραφές	159
7.2.1.3	Εγκεκριμένες Ηλεκτρονικές Υπογραφές	160
7.2.1.4	Εγκεκριμένες Διατάξεις Δημιουργίας Ηλεκτρονικής Υπογραφής	161
7.2.2	Νομικός Ορισμός των Εγκεκριμένων Ηλεκτρονικών Υπογραφών	162
7.2.2.1	Διεύρυνση Υπογραφών	162
7.2.2.2	Πρότυπα Σχετικά με τις Εγκεκριμένες Ηλεκτρονικές Υπογραφές	163
7.2.2.3	Εφαρμογές Δημιουργίας Υπογραφών	164
7.2.3	Ηλεκτρονικές Σφραγίδες	165
7.2.4	Ηλεκτρονικές Χρονοσφραγίδες	165
7.2.5	Ηλεκτρονικές Υπηρεσίες Συστημάτων Παράδοσης	166
7.2.6	Πιστοποίηση Γνησιότητας Ιστοτόπων	167
7.2.7	Κατάλογοι Εμπιστοσύνης της ΕΕ	168
7.3	Ψηφιακά Πορτοφόλια	168
7.3.1	Ευρωπαϊκό Ψηφιακό Πορτοφόλι	169
8	Μηχανισμοί Ενίσχυσης του Απορρήτου	173
8.1	Ασφαλείς Υπολογισμοί Πολλαπλών Οντοτήτων	173
8.1.1	Το Πρωτόκολλο του Yao (Garbled Circuit)	174
8.1.1.1	Δημιουργία Λογικού Κυκλώματος	175
8.1.1.2	Κρυπτογράφηση Κυκλώματος	175
8.1.1.3	Μεταφορά Δεδομένων	176
8.1.1.4	Αποκρυπτογράφηση Κυκλώματος	176
8.1.1.5	Αποκάλυψη της Εξόδου	177
8.1.2	Άλλα Πρωτόκολλα MPC	177
8.2	Ομομορφική Κρυπτογράφηση	177
8.2.1	Μερική Ομομορφική Κρυπτογράφηση	179

8.2.2	Πλήρης Ομομορφική Κρυπτογράφηση	180
8.2.2.1	Πρώτη Γενιά FHE	180
8.2.2.2	Δεύτερη Γενιά FHE	181
8.2.2.3	Τρίτη Γενιά FHE	181
8.2.2.4	Τέταρτη Γενιά FHE	182
8.3	Αποδείξεις Μηδενικής Γνώσης	182
8.3.1	Το Πρωτόκολλο του Schnorr	183
8.3.2	Το Πρωτόκολλο Chaum-Pedersen	184
8.4	Ψηφιακές Υπογραφές Ενίσχυσης του Απορρήτου	185
8.4.1	Τυφλές Υπογραφές	185
8.4.2	Ομαδικές Υπογραφές	186
8.4.3	Υπογραφές Δακτυλίου	187
8.5	Κρυπτογράφηση Βάσει Ταυτότητας και Χαρακτηριστικών	188
8.5.1	Κρυπτογράφηση Βάσει Ταυτότητας	188
8.5.2	Κρυπτογράφηση Βάσει Χαρακτηριστικών	189
8.6	Ασκήσεις-Εργασίες	190
	Ασκήσεις	190
	Εργασίες	191

III Εφαρμογές 199

9	Προστασία Δεδομένων σε Κατάσταση Ηρεμίας	201
9.1	Δεδομένα σε Κατάσταση Ηρεμίας	201
9.2	Κρυπτογράφηση Δεδομένων σε Επίπεδο Δίσκου	203
9.2.1	Μέθοδοι Κρυπτογράφησης Δίσκου	206
9.2.1.1	Κλασικοί Τρόποι Λειτουργίας και Βελτιώσεις	207
9.2.1.2	Ειδικές Προσεγγίσεις Τρόπων Λειτουργίας	208
9.2.2	BitLocker	210
9.3	Κρυπτογράφηση Δεδομένων σε Επίπεδο Συστήματος Αρχείων	212
9.3.1	Σύστημα Κρυπτογράφησης Αρχείων (EFS)	213
9.4	Κρυπτογράφηση Δεδομένων σε Επίπεδο Βάσης Δεδομένων	215
9.4.1	Επίπεδο Κρυπτογράφησης	216
9.4.2	Αλγόριθμοι Κρυπτογράφησης και Τρόποι Λειτουργίας	217
9.4.3	Διαχείριση Κλειδιών Κρυπτογράφησης	218
9.4.4	Εφαρμογές σε Συστήματα Διαχείρισης Βάσεων Δεδομένων	218
9.4.5	Επιπτώσεις στην Απόδοση από την Χρήση Κρυπτογράφησης	219
9.4.6	Κρυπτογράφηση των Αντιγράφων Ασφαλείας	220
10	Προστασία Δεδομένων κατά την Μεταφορά	225
10.1	Εισαγωγή	225
10.2	Transport Layer Security	228
10.2.1	Ιστορική Αναδρομή	228
10.2.2	Πρωτόκολλο TLS 1.3	229
10.2.2.1	Πρωτόκολλο Χειραψίας TLS	229
10.2.2.2	Πρωτόκολλο Εγγραφής TLS	234
10.3	IPSec	235
10.3.1	Λειτουργία Σήραγγας	237
10.3.2	Λειτουργία Μεταφοράς	239

10.3.3	Συνχετισμοί Ασφαλείας	239
10.3.4	Συνδυασμοί Συνχετισμών Ασφαλείας	241
10.3.4.1	Παράδειγμα 1: Εφαρμογή του IPSec από Άκρο-σε-Άκρο	242
10.3.4.2	Παράδειγμα 2: Εφαρμογή του IPSec από Πύλη-σε-Πύλη	242
10.3.4.3	Παράδειγμα 3: Συνδυασμός των Παραδειγμάτων 1 και 2	242
10.3.4.4	Παράδειγμα 4: Υποστήριξη Απομακρυσμένου Σταθμού	243
10.4	Secure Shell Protocol	243
10.4.1	Τρόπος Λειτουργίας του SSH	244
10.4.1.1	Διαπραγμάτευση Έκδοσης	244
10.4.1.2	Διαπραγμάτευση Αλγορίθμου	244
10.4.1.3	Ανταλλαγή Κλειδιών	246
10.4.1.4	Αυθεντικοποίηση Πελάτη	246
11	Αλυσίδες μπλοκ	249
11.1	Βασικές Έννοιες	250
11.1.1	Τεχνολογία Κατανεμημένου Καθολικού	250
11.1.2	Τεχνολογία Αλυσίδας Μπλοκ	250
11.1.3	Μπλοκ	250
11.1.4	Συναλλαγές	252
11.2	Κατηγοριοποίηση των Αλυσίδων Μπλοκ	252
11.2.1	Αλυσίδες Μπλοκ με Άδεια	253
11.2.2	Αλυσίδες Μπλοκ χωρίς Άδεια	254
11.2.3	Δικαιώματα και Διαφορές	254
11.3	Βασικά Χαρακτηριστικά των Αλυσίδων Μπλοκ	255
11.3.1	Συναρτήσεις Σύνοψης	255
11.3.2	Κρυπτογραφία Δημοσίου Κλειδιού	256
11.3.2.1	Ψηφιακές Υπογραφές	256
11.3.3	Διευθύνσεις	256
11.3.4	Έξυπνα Συμβόλαια	257
11.4	Αλγόριθμος Συναίνεσης	257
11.4.1	Απόδειξη Εργασίας	257
11.4.2	Απόδειξη Συμμετοχής	258
11.4.3	(Πρακτική) Βυζαντινή Ανοχή Σφαλμάτων	259
11.4.4	Απόδειξη Αρχής	260
11.5	Δημοφιλείς Πλατφόρμες Αλυσίδων Μπλοκ	260
11.5.1	Bitcoin	260
11.5.2	Ethereum	262
11.5.3	Hyperledger Fabric	263
11.6	Προκλήσεις Ασφαλείας και Λειτουργίας των Αλυσίδων Μπλοκ	265
11.6.1	Επίθεση Sybil	265
11.6.2	Η Επίθεση του 51%	266
11.6.3	Εγωιστική Εξόρυξη	266
11.6.4	Διακλάδωση Αλυσίδας Μπλοκ	267
11.6.5	Καθυστέρηση Συναίνεσης	267
11.6.6	Διπλή Δαπάνη	268
11.6.7	Παρωχημένα και Ορφανά Μπλοκ	269
12	Διασφάλιση Απορρήτου και Ιδιωτικότητας	271
12.1	Απόρρητο και Ιδιωτικότητα	272

12.2 Δίκτυα Ανωνυμίας	273
12.2.1 Δίκτυα Μίξης (Mix-Nets)	273
12.2.2 Δρομολόγησης Onion Routing	275
12.2.2.1 The Onion Router (TOR)	275
12.2.2.2 Invisible Internet Project (I2P)	276
12.3 Ηλεκτρονική Ψηφοφορία	276
12.3.1 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Δικτύων Μίξης	276
12.3.2 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Ομομορφικής Κρυπτογράφησης	278
12.3.3 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Τυφλών Υπογραφών	278
12.3.4 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Αλυσίδων Μπλοκ	279
12.3.5 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Μετα-Κβαντικής Κρυπτογραφίας	280
12.4 Δημοπρασίες με Διασφάλιση Ιδιωτικότητας	281
12.4.1 Δημοπρασίες Πρώτης Τιμής με Σφραγισμένη Προσφορά	281
12.4.2 Δημοπρασίες Δεύτερης Τιμής με Σφραγισμένη Προσφορά	283
12.5 Ιδιωτική Ανάκτηση Πληροφοριών	284
12.5.1 Σχήματα PIR Ενός Μόνο Διακομιστή	285
12.5.2 Σχήματα PIR Πολλαπλών Διακομιστών	285
12.5.2.1 Σχήμα PIR του Chor	286
12.5.2.2 Σχήμα PIR του Goldberg	286
12.6 Εξόρυξη Δεδομένων με Διασφάλιση Ιδιωτικότητας	287
12.6.1 Λύσεις PPDM με Χρήση Ασφαλών Υπολογισμών Πολλαπλών Οντοτήτων	289
12.6.2 Λύσεις PPDM με Κοινή Χρήση Μυστικών	290
12.6.3 Λύσεις PPDM με Χρήση Ομομορφικής Κρυπτογράφησης	291
12.7 Μηχανική Μάθηση με Διασφάλιση Ιδιωτικότητας	292
12.7.1 Λύσεις PPDL με Χρήση Ομομορφικής Κρυπτογράφησης	293
12.7.2 Λύσεις PPDL με Χρήση Ασφαλών Υπολογισμών Πολλαπλών Οντοτήτων	295
IV Ειδικά Θέματα	309
13 Ελαφρά Κρυπτογραφία	311
13.1 Εισαγωγή	311
13.2 Κατηγορίες και Χαρακτηριστικά	312
13.3 Αλγόριθμοι Ελαφράς Κρυπτογραφίας	313
13.3.1 Αλγόριθμος ASCON	314
13.3.1.1 Αντιμετάθεση ASCON	315
13.3.1.2 Λειτουργίες Πιστοποιημένης Κρυπτογράφησης ASCON	316
13.3.1.3 Λειτουργίες Συναρτήσεων Σύνοψης ASCON	317
13.3.2 Αλγόριθμος GIFT-COFB	318
13.3.2.1 Λειτουργία Πιστοποιημένης Κρυπτογράφησης GIFT-COFB	319
13.3.2.2 Δομικά Στοιχεία του COFB	320
13.3.2.3 Δομικά Στοιχεία του GIFT	320
13.3.3 Αλγόριθμος SPARKLE (SCHWAEMM και ESCH)	322
13.3.3.1 Αντιμετάθεση SPARKLE	322
13.3.3.2 Λειτουργίες Πιστοποιημένης Κρυπτογράφησης SCHWAEMM	323
13.3.3.3 Λειτουργίες Συναρτήσεων Σύνοψης ESCH	324
13.3.4 Αλγόριθμος TinyJAMBU	325
13.3.4.1 Αντιμετάθεση P_n με Χρήση Κλειδιού	326
13.3.4.2 Λειτουργίες Πιστοποιημένης Κρυπτογράφησης TinyJAMBU	327

13.3.5	Αλγόριθμος Xoodyak	328
13.3.5.1	Αντιμετάθεση Xoodoo	329
13.3.5.2	Τρόπος Λειτουργίας Cyclist	330
13.3.5.3	Λειτουργία Πιστοποιημένης Κρυπτογράφησης Xoodyak	331
13.3.5.4	Λειτουργία Συνάρτησης Σύνοψης Xoodyak	331
14	Εφαρμοσμένη Κβαντική Κρυπτογραφία	335
14.1	Ορισμός και Ιστορική Αναδρομή της Κβαντικής Κρυπτογραφίας	335
14.2	Κβαντική Θεωρία Πληροφοριών	336
14.2.1	Κβαντικά Bits	336
14.2.2	Γραμμικοί Τελεστές	338
14.2.3	Κβαντική Μέτρηση	339
14.2.4	Το Θεώρημα της Μη-Κλωνοποίησης	340
14.3	Κβαντική Διανομή Κλειδών	340
14.3.1	Το Πρωτόκολλο BB84	341
14.3.2	Το Πρωτόκολλο E91	343
14.4	Κβαντική Κοινή Χρήση Μυστικών	344
14.5	Κβαντική Ρίψη Νομίσματος	345
14.5.1	Ρίψη Νομίσματος με Χρήση Συζευγμένης Κωδικοποίησης	347
14.5.2	Το Πρωτόκολλο Dip Dip Boom	348
14.6	Κβαντική Δέσμευση	349
14.7	Οριοθετημένο και Θορυβώδες Μοντέλο Κβαντικής Αποθήκευσης	350
14.8	Κβαντική Κρυπτογραφία βάσει Θέσης	350
15	Μετα-Κβαντική Κρυπτογραφία	355
15.1	Εισαγωγή	355
15.2	Μετα-Κβαντική Κρυπτογραφία	356
15.3	Οικογένειες Μετα-Κβαντικών Αλγορίθμων	357
15.3.1	Κρυπτογραφία Βασισμένη σε Συναρτήσεις Σύνοψης	358
15.3.1.1	Υπογραφές μιας Χρήσης	359
15.3.1.2	Από Υπογραφές μιας Χρήσης σε Υπογραφές Πολλαπλών Χρήσεων	363
15.3.1.3	SPHINCS	364
15.3.1.4	SPHINCS+	369
15.3.2	Κρυπτογραφία Βασισμένη σε Πλέγματα	370
15.3.2.1	Προβλήματα Σχετικά με τη Θεωρία των Πλεγμάτων	371
15.3.3	Κρυπτογραφία Βασισμένη σε Κώδικα	372
15.3.4	Κρυπτογραφία Βασισμένη σε Ισογένειες	373
15.3.5	Πολυμεταβλητή Κρυπτογραφία	374
15.4	Μετάβαση στην Μετα-Κβαντική Εποχή	375
15.4.1	Υβριδικά Συστήματα	375
15.4.2	Μέτρα Προστασίας για Προ-Κβαντική Κρυπτογραφία	376
V	Παραρτήματα	381
A	Οδηγίες Χρήσης του CrypTool 2	383
A.1	CrypTool 2	383
A.1.1	Περιβάλλον Εργασίας	383
A.1.2	Το εργαλείο Wizard	383

A.1.3	To Workspace	383
A.1.4	Παράδειγμα Δημιουργίας Σεναρίου με το CrypTool 2	386
B	Απαντήσεις Ερωτήσεων - Λύσεις Ασκήσεων	389
B.1	Κεφάλαιο 1	389
B.1.1	Λύσεις Εργασιών	389
B.2	Κεφάλαιο 2	389
B.2.1	Λύσεις Ασκήσεων	389
B.2.2	Λύσεις Εργασιών	391
B.3	Κεφάλαιο 3	392
B.3.1	Λύσεις Ασκήσεων	392
B.3.2	Λύσεις Εργασιών	393
B.4	Κεφάλαιο 4	393
B.4.1	Λύσεις Ασκήσεων	393
B.4.2	Λύσεις Εργασιών	394
B.5	Κεφάλαιο 5	395
B.5.1	Λύσεις Εργασιών	395
B.6	Κεφάλαιο 8	396
B.6.1	Λύσεις Ασκήσεων	396
B.6.2	Λύσεις Εργασιών	397

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

1.1	Η βασική λειτουργία κρυπτογράφησης-αποκρυπτογράφησης.	4
1.2	Κρυπτογράφηση κατά μπλοκ.	6
1.3	Ο κώδικας Feistel.	9
1.4	Ο αλγόριθμος AES.	10
1.5	Τρόπος λειτουργίας ECB.	12
1.6	Τρόπος λειτουργίας CBC.	14
1.7	Κρυπτογράφηση με τρόπο λειτουργίας ECB και CBC	15
1.8	Κρυπτογράφηση με τρόπο λειτουργίας CTR.	15
1.9	Αποκρυπτογράφηση με τρόπο λειτουργίας CTR.	16
1.10	Κρυπτογράφηση με τρόπο λειτουργίας CFB.	17
1.11	Αποκρυπτογράφηση με τρόπο λειτουργίας CFB.	17
1.12	Κρυπτογράφηση με τρόπο λειτουργίας OFB.	17
1.13	Αποκρυπτογράφηση με τρόπο λειτουργίας OFB.	18
1.14	Αλγόριθμος κρυπτογράφησης ροής.	18
1.15	Κρυπτογράφηση κειμένου με την χρήση του αλγορίθμου AES στο CrypTool 2.	26
2.1	Διαδικασία κρυπτογράφησης και αποκρυπτογράφησης στα κρυπτοσυστήματα δημοσίου κλειδιού.	30
2.2	Μεγέθη RSA κλειδιών που έχουν παραγοντοποιηθεί από το 1991 μέχρι και το 2020.	33
2.3	Διαδικασία κρυπτογράφησης και αποκρυπτογράφησης του κωδικού πρόσβασης με βάση το κρυπτοσύστημα RSA.	46
3.1	Συνάρτηση σύνοψης.	52
3.2	Κατασκευή Merkle-Damgård.	54
3.3	Κατασκευή Wide-Pipe.	54
3.4	Κατασκευή HAIFA.	55
3.5	Κατασκευή Sponge.	56
3.6	Συναρτήσεις συμπίεσης που βασίζονται σε κρυπτογράφηση μπλοκ.	57
3.7	Παράδειγμα ενός δυαδικού δένδρου Merkle με τέσσερα μπλοκ δεδομένων.	58
3.8	Λειτουργία της συνάρτησης σύνοψης MDS.	59
3.9	Λειτουργία της συνάρτησης σύνοψης Whirlpool.	60

3.10 Λειτουργία της συνάρτησης σύνοψης SHA-1.	63
3.11 Λειτουργία της συνάρτησης σύνοψης SHA-2.	64
3.12 Λειτουργία της συνάρτησης σύνοψης SHA-3.	66
3.13 Συγκριτική παρουσίαση ταχύτητας της οικογένεια συναρτήσεων BLAKE για δεδομένα εισόδου 16 KB σε έναν σύγχρονο εξυπηρετητή (επεξεργαστής Cascade Lake-SP 8275CL).	67
3.14 Λειτουργία της συνάρτησης σύνοψης BLAKE.	68
4.1 Αρχιτεκτονικό μοντέλο μιας γεννήτριας ψευδοτυχαίων αριθμών.	80
4.2 Εξέλιξη της κατάστασης $S = (V, C, cnt)$ μετά από μια κλήση του μηχανισμού HASH-DRBG.	85
4.3 Εξέλιξη της κατάστασης $S = (K, V, cnt)$ μετά από μια κλήση του μηχανισμού HMAC-DRBG.	87
4.4 Εξέλιξη της κατάστασης $S = (K, V, cnt)$ μετά από μια κλήση του μηχανισμού CTR-DRBG.	89
5.1 Οι δύο περιπτώσεις δημιουργίας του CMAC.	102
5.2 Αυθεντικοποιημένη κρυπτογράφηση με την μέθοδο κρυπτογράφηση και μετά MAC.	107
5.3 Αλγόριθμος αυθεντικοποιημένης κρυπτογράφησης GCM.	110
5.4 Υπολογισμός της συνάρτησης $GHASH_H(X_1\ X_2\ \dots\ X_m) = Y_m$	111
5.5 Υπολογισμός της συνάρτησης $GCTR_K(ICB, X_1\ X_2\ \dots\ X_n^*) = Y_1\ Y_2\ \dots\ Y_n^*$	112
5.6 Βασικά σχήματα υπογραφών.	114
5.7 Κρυπτογράφηση συμμετρικού κλειδιού με τη χρήση κρυπτογραφίας δημοσίου κλειδιού.	119
5.8 Γεννήτρια κλειδιών RSA.	119
6.1 Κύκλος ζωής κρυπτογραφικών κλειδιών κατά NIST (NIST 800-57 [1]).	126
6.2 Επισκόπηση του πρωτοκόλλου Kerberos.	136
6.3 Ανταλλαγή μηνυμάτων στο πρωτόκολλο Kerberos.	137
6.4 Επισκόπηση του πρωτοκόλλου Kerberos με σύνδεση διαφορετικών τομέων διαχείρισης.	138
6.5 Πρωτόκολλο Needham-Schroeder.	139
7.1 Ιεραρχικό μοντέλο ΥΔΚ.	151
7.2 Παράδειγμα ψηφιακού πιστοποιητικού Αρχής Πιστοποίησης Ρίζας.	154
7.3 Παράδειγμα Λίστας Ανακληθέντων Πιστοποιητικών.	156
7.4 Ενωσιακό σήμα εμπιστοσύνης για τις εγκεκριμένες υπηρεσίες εμπιστοσύνης.	159
7.5 Οι εγκεκριμένες ηλεκτρονικές υπογραφές έχουν το ισοδύναμο νομικό αποτέλεσμα με τις χειρόγραφες υπογραφές.	162
7.6 Διαδικασία δημιουργίας χρονοσφραγίδας.	166
7.7 Βασική λειτουργικότητα ψηφιακού πορτοφολιού.	170
8.1 Παράδειγμα κρυπτογράφησης κυκλώματος για την λογική πύλη AND.	176
8.2 Αποτύπωση της χρησιμότητας, της προστασίας και της απόδοσης στην ομομορφική κρυπτογράφηση.	178
8.3 Τα βήματα εκτέλεσης του πρωτοκόλλου του Schnorr.	184
8.4 Τα βήματα εκτέλεσης του πρωτοκόλλου του Chaum-Pedersen.	184
8.5 Η δακτυλιοειδής δομή του αλγορίθμου δημιουργίας υπογραφής δακτυλίου.	188
8.6 Επισκόπηση των βημάτων κρυπτογράφησης βάσει ταυτότητας.	189
8.7 Η θέση της μαγικής πόρτας μεταξύ των σημείων 3 και 4 στην σπηλιά που μπορεί να ανοίξει μόνο με την χρήση ενός μυστικού κλειδιού.	191
9.1 Οι τρεις καταστάσεις των ψηφιακών δεδομένων.	202
9.2 Τρόπος λειτουργίας XEX κατά την κρυπτογράφηση.	209
9.3 Τρόπος λειτουργίας XTS κατά την κρυπτογράφηση.	210
9.4 Λειτουργία του συστήματος κρυπτογράφησης αρχείων (EFS).	214
9.5 Τρεις επιλογές για το επίπεδο κρυπτογράφησης μιας βάσης δεδομένων.	217

9.6 Διαφορετικές προσεγγίσεις διαχείρισης κρυπτογραφικών κλειδιών.	219
10.1 Πρωτόκολλα ασφαλείας στην στοίβα TCP/IP.	226
10.2 Πρωτόκολλο χειραψίας στο TLS 1.3.	231
10.3 Σήραγγα 1ης φάσης IKE.	236
10.4 Σήραγγα 2ης φάσης IKE.	236
10.5 Ανταλλαγή δεδομένων με τη χρήση σήραγγας 2ης φάσης IKE.	236
10.6 Χρήση του IPSec μεταξύ δύο πυλών, με τη χρήση της λειτουργίας σήραγγας.	238
10.7 AH και ESP με χρήση λειτουργίας σήραγγας.	238
10.8 Χρήση του IPSec σε λειτουργία μεταφοράς.	239
10.9 AH και ESP με χρήση λειτουργίας μεταφοράς.	240
10.10 IPSec από άκρο-σε-άκρο (υλοποίηση σε σταθμούς).	242
10.11 IPSec μεταξύ δύο πυλών.	243
10.12 IPSec μεταξύ δύο πυλών και σταθμών.	243
10.13 Υποστήριξη απομακρυσμένου σταθμού.	243
10.14 Οι φάσεις λειτουργίας του SSH.	245
11.1 Πολυεπίπεδη αρχιτεκτονική αλυσίδας μπλοκ.	251
11.2 Δομή των μπλοκ σε μια αλυσίδα μπλοκ.	251
11.3 Αναπαραστάσεις δικτύων αλυσίδας μπλοκ με άδεια.	253
11.4 Αναπαράσταση ενός δημόσιου δικτύου αλυσίδας μπλοκ χωρίς άδεια.	254
11.5 Επισκόπηση του συστήματος Bitcoin.	261
11.6 Επισκόπηση του τρόπου λειτουργίας του Ethereum.	263
11.7 Επισκόπηση του συστήματος Hyperledger Fabric.	264
11.8 Δημιουργία διακλάδωσης σε μια αλυσίδα μπλοκ.	267
11.9 Παράδειγμα παρωχημένων και ορφανών μπλοκ.	269
12.1 Παράδειγμα επικοινωνίας σε ένα δίκτυο μίξης (Mix-Net).	274
12.2 Παράδειγμα επικοινωνίας στο δίκτυο TOR.	275
12.3 Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση δικτύων μίξης.	277
12.4 Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση ομομορφικής κρυπτογράφησης.	278
12.5 Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση τυφλών υπογραφών.	279
12.6 Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση αλυσίδων μπλοκ.	280
12.7 Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση μετα-κβαντικής κρυπτογραφίας.	281
12.8 Γενική αρχιτεκτονική μιας δημοπρασίας πρώτης τιμής με σφραγισμένη προσφορά.	282
12.9 Πιθανά σενάρια κατόχων δεδομένων στην εξόρυξη δεδομένων.	288
12.10 Απεικόνιση ενός σχήματος βαθιάς μάθησης με διασφάλιση ιδιωτικότητας (PPDL).	293
12.11 Γενική δομή σχημάτων PPDL που βασίζονται σε ομομορφική κρυπτογράφηση.	294
12.12 Γενική δομή σχημάτων PPDL κατά την φάση εκπαίδευσης που βασίζονται σε ασφαλείς υπολογισμούς πολλαπλών οντοτήτων (MPC).	296
12.13 Γενική δομή σχημάτων PPDL κατά την φάση εξαγωγής συμπερασμάτων που βασίζονται σε ασφαλείς υπολογισμούς δύο οντοτήτων (2PC).	297
13.1 Χρόνος εκτέλεσης των πιο γρήγορων υλοποίησεων των κύριων παραλλαγών AEAD για πιστοποιημένη κρυπτογράφηση σε διάφορους μικροελεγκτές.	314
13.2 Γραμμικός και μη γραμμικός μετασχηματισμός της αντιμετάθεσης ASCON.	315
13.3 Διαδικασία κρυπτογράφησης στην κρυπτογραφική σουίτα ASCON.	316
13.4 Διαδικασία αποκρυπτογράφησης στην κρυπτογραφική σουίτα ASCON.	317

13.5 Διαδικασία δημιουργίας σύνοψης στην κρυπτογραφική σουίτα ASCON.	318
13.6 Διαδικασία κρυπτογράφησης στον αλγόριθμο GIFT-COFB.	319
13.7 Συνολική δομή ενός γύρου SPARKLE και του αντίστοιχου ARX-box Alzette A_c	323
13.8 Διαδικασία κρυπτογράφησης του κρυπτογραφικού σχήματος SCHWAEMM256-128.	324
13.9 Διαδικασία δημιουργίας σύνοψης του κρυπτογραφικού σχήματος ESCH.	325
13.10 Μη γραμμικός καταχωρητής NFSR 128-bit στο κρυπτογραφικό σχήμα TinyJAMBU.	326
13.11 Διαδικασία κρυπτογράφησης του κρυπτογραφικού σχήματος TinyJAMBU.	327
13.12 Εσωτερική κατάσταση της αντιμετάθεσης Xoodoo.	330
14.1 Αναπαράσταση ενός qubit στην σφαίρα Bloch.	337
14.2 Επίδειξη του πρωτοκόλλου της κβαντικής ρίψης νομίσματος.	348
15.1 Δημιουργία κλειδιών στο σύστημα Lamport.	360
15.2 Δημιουργία υπογραφής στη σύνοψη M με το σύστημα Lamport. ;	360
15.3 Δημιουργία κλειδιών και υπογραφής και επαλήθευση υπογραφής με το σύστημα Winter- nitz.	362
15.4 Σύνθεση των συστημάτων υπογραφών που βασίζονται σε συνόψεις.	364
15.5 Δομή δέντρου του SPHINCS (γενική αναπαράσταση).	365
15.6 Δομή δέντρου του SPHINCS (σχηματική αναπαράσταση).	366
15.7 Πλέγμα που σχηματίζεται από δύο δισδιάστατα διανύσματα.	371
A.1 Η πρώτη εικόνα του προγράμματος κατά την έναρξη του CrypTool 2.	384
A.2 Επιλέγοντας αλγόριθμο κρυπτογράφησης στο CrypTool 2 με το εργαλείο Wizard.	384
A.3 Το περιβάλλον εργασίας του CrypTool 2.	385
A.4 Εργαλειοθήκη διαθέσιμων επιλογών του CrypTool 2.	385
A.5 Μενού διαθέσιμων επιλογών ενός στοιχείου (επιλογή δεξιά κλικ πάνω στο στοιχείο).	386
A.6 Εύρεση στοιχείων σχετικών με τη λέξη “text”.	386
A.7 Εισαγωγή του στοιχείου “Text Input” στο περιβάλλον εργασίας.	387
A.8 Εισαγωγή των στοιχείων “SHA” και “Text Output”.	387
A.9 Διασύνδεση όλων των στοιχείων.	387
B.1 Γεννήτρια κλειδιών RSA στο περιβάλλον CrypTool 2.	391

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

1.1	Κρυπτογραφικοί αλγόριθμοι μπλοκ.	7
1.2	Χαρακτηριστικά του αλγορίθμου AES.	10
1.3	Τρόποι λειτουργίας αλγορίθμων μπλοκ.	12
1.4	Αλγόριθμοι κρυπτογράφησης ροής.	19
2.1	Συγκριτική παρουσίαση μεταξύ του μεγέθους κλειδιών RSA και ECC.	40
3.1	Συγκριτική παρουσίαση της οικογένειας συναρτήσεων σύνοψης SHA.	61
3.2	Συγκριτική παρουσίαση της οικογένειας συναρτήσεων σύνοψης BLAKE.	67
4.1	Παράμετροι του μηχανισμού HASH-DRBG για την οικογένεια συναρτήσεων σύνοψης SHA-2.	85
4.2	Παράμετροι του μηχανισμού CTR-DRBG (όπου $B = (2^{ctr_len} - 4) \cdot blocklen$).	89
5.1	Σχήματα MAC.	99
5.2	Οι συναρτήσεις MAC σύμφωνα με το ISO 9797-1.	101
5.3	Αυθεντικοποιημένη κρυπτογράφηση.	106
5.4	Συστήματα ψηφιακών υπογραφών βασισμένα σε κρυπτογραφία δημοσίου κλειδιού.	113
7.1	Τύποι και μεγέθη κλειδιών.	157
7.2	Συγκριτικός πίνακας λειτουργιών που προσφέρονται από τους διάφορους τύπους εγκεκριμένων υπηρεσιών εμπιστοσύνης.	159
8.1	Συγκριτική παρουσίαση γνωστών πρωτοκόλλων MPC με παθητικούς αντιπάλους.	177
8.2	Ιστορική αναδρομή της ομομορφικής κρυπτογράφησης.	179
13.1	Κατηγορίες υλοποίησης υλικού με την μονάδα μέτρησης Gate Equivalent (GE).	313
13.2	Συγκριτική παρουσίαση των φιναλίστ αλγορίθμων ελαφράς κρυπτογραφίας του NIST.	314
13.3	Παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης ASCON.	317
13.4	Παράμετροι για τις συναρτήσεις σύνοψης ASCON.	318
13.5	Διαφορετικές εκδόσεις της αντιμετάθεσης SPARKLE.	323
13.6	Παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης SCHWAEMM.	324
13.7	Παράμετροι για τις συναρτήσεις σύνοψης ESCH.	326

13.8 Παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης TinyJAMBU.	328
14.1 Κατάσταση πόλωσης φωτονίων ανάλογα με την τιμή και τη βάση ενός bit.	342
14.2 Παράδειγμα εκτέλεσης του πρωτοκόλλου BB84.	343
14.3 Η επίδραση των μετρήσεων της Αλίκης και του Μπάμπη στην κατάσταση του Τσάρλι για την τριπλέτα GHZ.	346
15.1 Συναρτήσεις που χρησιμοποιούνται στο SPHINCS.	365

ΠΡΟΛΟΓΟΣ

ΓΕΩΡΓΙΟΣ ΔΡΟΣΑΤΟΣ
ΙΩΑΝΝΗΣ ΜΑΥΡΙΔΗΣ
ΚΩΝΣΤΑΝΤΙΝΟΣ ΡΑΝΤΟΣ

Το σύγγραμμα «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας» παρουσιάζει και αναλύει σύγχρονους κρυπτογραφικούς αλγορίθμους, μηχανισμούς και εφαρμογές αυτών που συμβάλλουν στην προστασία της εμπιστευτικότητας και της ακεραιότητας, στην αυθεντικοποίησης και στη διαχείριση κλειδιών. Μέσα από τα θέματα που καλύπτονται στο σύγγραμμα ο αναγνώστης έχει τη δυνατότητα να κατανοήσει πλήρως και σε βάθος τα ζητήματα εφαρμογής των βασικών στοιχείων της κρυπτογραφίας, και να γνωρίσει νέες τάσεις και καινοτόμα κρυπτοσυστήματα καθώς και τον τρόπο λειτουργίας ορισμένων διαθέσιμων αλγορίθμων και μηχανισμών.

Το σύγγραμμα περιλαμβάνει 15 αυτοτελή κεφάλαια από τα οποία οι διδάσκοντες μπορούν να επιλέξουν αυτά που είναι πιο συναφή στις θεματικές ενότητες του εκάστοτε μαθήματος. Επιπλέον, τα εισαγωγικά κεφάλαια συμπληρώνονται από εργαστηριακές ασκήσεις και πρόσθετο διαδραστικό εκπαιδευτικό υλικό με την μορφή επίδειξης (demo), τόσο στο εκπαιδευτικό περιβάλλον CryptTool2 όσο και στην γλώσσα προγραμματισμού Java κάνοντας χρήση του περιβάλλοντος ανάπτυξης Eclipse.

Στο πλαίσιο αυτό, το σύγγραμμα ξεκινά με την παρουσίαση των σύγχρονων κρυπτογραφικών αλγορίθμων και μηχανισμών αναλύοντας τα βασικά στοιχεία της κρυπτογραφίας, όπως είναι η συμμετρική και ασύμμετρη κρυπτογραφία και οι συναρτήσεις σύνοψης, και στη συνέχεια παρουσιάζει βασικούς κρυπτογραφικούς μηχανισμούς, όπως είναι οι γεννήτριες τυχαίων αριθμών, και οι μηχανισμοί ακεραιότητας και αυθεντικοποίησης δεδομένων, αναλύει θέματα ορθής διαχείρισης κρυπτογραφικών κλειδιών, υποδομών δημοσίου κλειδιού, υπηρεσιών εμπιστοσύνης, και μηχανισμών ενίσχυσης του απορρήτου. Βάσει αυτών, αναλύονται οριζόντιες εφαρμογές της κρυπτογραφίας που περιλαμβάνουν την προστασία δεδομένων σε κατάσταση ηρεμίας και κατά την μεταφορά, αλυσίδες μπλοκ καθώς και λύσεις διασφάλισης απορρήτου και ιδιωτικότητας. Στο τελευταίο μέρος του συγγράμματος αναλύονται ειδικά θέματα της κρυπτογραφίας που αφορούν την ελαφρά κρυπτογραφία, την κβαντική και την μετα-κβαντική κρυπτογραφία.

Συνολικά, το σύγγραμμα αποσκοπεί στο να καλύψει τις ανάγκες που υπάρχουν σε πολλά μαθήματα που πραγματεύονται ζητήματα Ασφάλειας Πληροφοριών και Ιδιωτικότητας. Καθώς οι λύσεις σε πολλά από αυτά τα ζητήματα εδράζονται στην αξιοποίηση της Κρυπτογραφίας, προβάλλει ξεκάθαρα την ανάγκη για μελέτη των διαφόρων μορφών και εκφάνσεών της, με σκοπό την αξιολόγηση των εναλλακτικών σεναρίων και δια-

δικασιών υλοποίησης. Η σφαιρική αλλά χωρίς υπερβολικές εμβαθύνσεις παρουσίαση του θεωρητικού υπόβαθρου σε συνδυασμό με την απλουστευμένη αλλά κριτική και συστηματική παρουσίαση πρακτικών παραδειγμάτων εφαρμογής της κρυπτογραφίας, αποτελεί έναν σημαντικό παράγοντα για την αξιοποίηση του παρόντος συγγράμματος από έναν σημαντικό αριθμό μαθημάτων ασφάλειας πληροφοριών και ιδιωτικότητας που εντοπίζονται σχεδόν σε όλα τα προγράμματα σπουδών πληροφορικής προπτυχιακού και μεταπτυχιακού επιπέδου. Τέλος, με την παρουσίαση νέων τάσεων και ειδικών θεμάτων σχετικών με την εξέλιξη αλλά και την εφαρμογή της κρυπτογραφίας σε αναδυόμενες περιοχές, το παρών σύγγραμμα μπορεί επίσης να λειτουργήσει συμπληρωματικά σε υπάρχοντα και αναγνωρισμένα συγγράμματα του χώρου.

Μέρος Ι

ΒΑΣΙΚΕΣ ΘΕΜΕΛΙΩΣΕΙΣ ΚΑΙ ΤΕΧΝΙΚΕΣ

ΚΕΦΑΛΑΙΟ 1

ΚΡΥΠΤΟΓΡΑΦΙΑ ΣΥΜΜΕΤΡΙΚΟΥ ΚΛΕΙΔΙΟΥ

Περίληψη

Η κρυπτογραφία συμμετρικού κλειδιού [1] αποτελεί ένα από βασικά στοιχεία στη δημιουργία αποτελεσμάτων κρυπτοσυστημάτων. Χρησιμοποιείται εκτενώς για την κρυπτογράφηση δεδομένων, εξασφαλίζοντας την εμπιστευτικότητα των μηνυμάτων, και αποτελεί τη βάση για τη δημιουργία άλλων μηχανισμών όπως κωδικών αυθεντικοποίησης μηνύματος για την παροχή ακεραιότητας και πιστοποίησης της πηγής των μηνυμάτων, γεννητριών ψευδοτυχαίων αριθμών και συναρτήσεων κατακερματισμού. Στο κεφάλαιο αυτό παρουσιάζονται οι βασικές αρχές και ιδιότητες των δύο κατηγοριών συμμετρικών κρυπτοσυστημάτων: των αλγορίθμων μπλοκ (block) και ροής (stream) στις Ενότητες 1.3 και 1.5 αντίστοιχα. Επιπλέον αναλύονται σύγχρονοι αλγόριθμοι που αποτελούν τη βάση δημιουργίας ισχυρών συστημάτων προστασίας δεδομένων, όπως είναι ο αλγόριθμος μπλοκ AES (Ενότητα 1.3.2) και ο αλγόριθμος ροής HC-128 (Ενότητα 1.5.1). Αναλύονται επίσης οι τρόποι λειτουργίας των αλγορίθμων μπλοκ ECB, CBC, CTR, OFB και CFB που συμβάλλουν στην αποτελεσματικότερη και ασφαλέστερη χρήση των αλγορίθμων (Ενότητα 1.4).

Προαπαιτούμενη γνώση: –.

1.1 Εισαγωγή

Η κρυπτογράφηση αποτελεί την κύρια εφαρμογή της κρυπτογραφίας. Καθιστά τα δεδομένα μη αναγνώσιμα και μη επεξεργάσιμα από ένα πληροφοριακό σύστημα προκειμένου να διασφαλιστεί η εμπιστευτικότητά τους. Η κρυπτογράφηση χρησιμοποιεί έναν αλγόριθμο που ονομάζεται αλγόριθμος κρυπτογράφησης (cryptographic algorithm ή cipher) και μια μυστική τιμή που ονομάζεται κρυπτογραφικό κλειδί ή απλά κλειδί (key). Η χρήση του αλγορίθμου υλοποιείται σε ένα κρυπτοσύστημα (cryptosystem).

Αυτό το κεφάλαιο θα επικεντρωθεί στη συμμετρική κρυπτογραφία, γνωστή και ως κρυπτογραφία συμμετρικού κλειδιού. Στη συμμετρική κρυπτογραφία, το κλειδί που χρησιμοποιείται για την αποκρυπτογράφηση είναι το ίδιο με το κλειδί που χρησιμοποιείται για την κρυπτογράφηση (σε αντίθεση με την ασύμμετρη κρυπτογραφία ή την κρυπτογραφία δημοσίου κλειδιού (βλέπε Κεφάλαιο 2), στην οποία το κλειδί που χρησιμο-

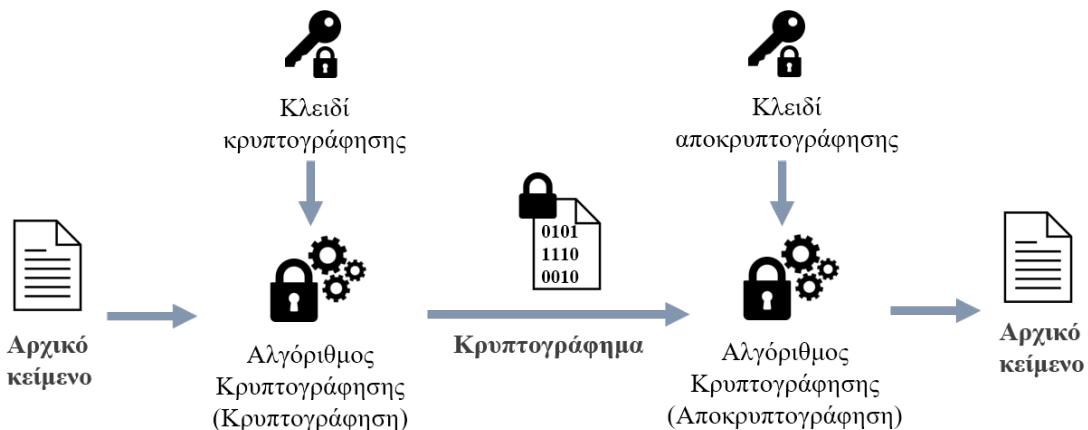
Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

ποιείται για την αποκρυπτογράφηση είναι διαφορετικό από το κλειδί που χρησιμοποιείται για την κρυπτογράφηση). Μια βασική αρχή της κρυπτογραφίας, γνωστή ως Αρχή του Kerckhoffs, είναι πως η ασφάλεια ενός κρυπτοσυστήματος θα πρέπει να παραμένει αικόμη και όταν οτιδήποτε σχετικό με αυτό, εκτός από το κλειδί, γίνει δημοσίως γνωστό. Με άλλα λόγια, η ασφάλεια ενός κρυπτοσυστήματος δεν πρέπει να βασίζεται στη μυστικότητα του κρυπτογραφικού αλγορίθμου αλλά στη μυστικότητα του κλειδιού αποκρυπτογράφησης. Για παράδειγμα, εάν δεν είναι γνωστό το μυστικό κλειδί αποκρυπτογράφησης, θα είναι αδύνατη η αποκρυπτογράφηση του κρυπτογραφημένου μηνύματος. Βασική προϋπόθεση ωστόσο, αποτελεί ή χρήση ισχυρών αλγορίθμων κρυπτογραφίας, η μελέτη των οποίων αποτελεί και αντικείμενο αυτού του κεφαλαίου, ξεκινώντας από κάποιες αδύναμες μορφές συμμετρικής κρυπτογραφίας οι οποίες ωστόσο αποτελούν τη βάση για τη δημιουργία ισχυρών αλγορίθμων.

1.2 Βασικές Έννοιες

Όταν κρυπτογραφούμε ένα μήνυμα, το αρχικό κείμενο (plaintext) αναφέρεται στο, καταληπτό από τον άνθρωπο ή επεξεργάσιμο από ένα πληροφοριακό σύστημα, μη κρυπτογραφημένο μήνυμα, το οποίο δίνεται ως είσοδος στον αλγόριθμο κρυπτογραφίας, ενώ το κρυπτοκείμενο (ciphertext) στο αποτέλεσμα της διαδικασίας κρυπτογράφησης, ένα κείμενο το οποίο δεν είναι καταληπτό από τον άνθρωπο ή ένα σύνολο δεδομένων μη επεξεργάσιμων από ένα πληροφοριακό σύστημα (με εξαίρεση τα κρυπτοσυστήματα ομομορφικής κρυπτογραφίας τα οποία αναλύονται σε επόμενα κεφάλαια). Ένα κρυπτοσύστημα τυπικά υλοποιεί δύο λειτουργίες: την κρυπτογράφηση (encryption) η οποία μετατρέπει το αρχικό κείμενο σε κρυπτογραφημένο και την αποκρυπτογράφηση (decryption) που ανακτά το αρχικό κείμενο από το κρυπτογραφημένο. Για παράδειγμα, στο Σχήμα 1.1 απεικονίζεται η διαδικασία της κρυπτογράφησης E , που αναπαρίσταται ως πλαίσιο που παίρνει ως είσοδο το αρχικό κείμενο P και ένα κλειδί K , και παράγει ένα κρυπτοκείμενο C , ως έξοδο. Η διαδικασία της κρυπτογράφησης είναι $C = E_K(P)$. Ομοίως, η λειτουργία της αποκρυπτογράφησης είναι $P = D'_K(C)$, όπου K' το κλειδί της αποκρυπτογράφησης. Στους συμμετρικούς αλγορίθμους το K' είναι ίδιο με το K ή δημιουργείται πολύ εύκολα από αυτό.



Σχήμα 1.1: Η βασική λειτουργία κρυπτογράφησης-αποκρυπτογράφησης.

1.2.1 Αντικατάσταση

Οι περισσότεροι κλασικοί κρυπτογραφικοί αλγόριθμοι λειτουργούν αντικαθιστώντας κάθε γράμμα του αρχικού κειμένου με ένα άλλο γράμμα – με άλλα λόγια, εκτελώντας μια αντικατάσταση (substitution). Η αντικατάσταση τυπικά αποτελεί μια αλλαγή ενός γράμματος ή συμβόλου με ένα άλλο εντός του ίδιου αλφαριθμού. ή συνόλου συμβόλων. Το αλφάριθμο ή το σύνολο συμβόλων μπορεί να ποικίλλει. Για παράδειγμα, αντί για το

αγγλικό αλφάβητο, θα μπορούσε να είναι το ελληνικό αλφάβητο. αντί για γράμματα, μπορεί να είναι λέξεις, ή αριθμοί ή ακολουθία από bits.

Ένας τύπος κρυπτογράφησης με αντικατάσταση είναι η μονοαλφαριθμητική αντικατάσταση (monoalphabetic substitution), όπου τα ίδια γράμματα του αρχικού κειμένου αντικαθίστανται πάντα από τα ίδια γράμματα στο κρυπτοκείμενο. Κάθε γράμμα του αρχικού κειμένου αντικαθίσταται από το ίδιο γράμμα του αλφαριθμητού αντικατάστασης, όπως αυτό προκύπτει από τη μετάθεση του αρχικού αλφαριθμητού με βάση την τιμή του κλειδιού. Λόγω αυτής της ιδιότητάς τους, οι αλγόριθμοι είναι ευάλωτοι σε επιθέσεις στατιστικής ανάλυσης συχνότητας εμφάνισης των γραμμάτων. Μια τέτοια επίθεση κρυπτανάλυσης περιλαμβάνει τη μέτρηση της συχνότητας της εμφάνισης των γραμμάτων στο κρυπτοκείμενο και τη σύγκρισή τους με τη συχνότητα εμφάνισης των γραμμάτων του αλφαριθμητού στη βιβλιογραφία της γλώσσας που χρησιμοποιείται.

Στους αλγορίθμους πολυαλφαριθμητικής αντικατάστασης (polyalphabetic substitution) χρησιμοποιούνται πολλά μοτίβα αντικατάστασης και τα ίδια γράμματα του αρχικού κειμένου κρυπτογραφούνται διαφορετικά με βάση τη θέση τους στο κείμενο. Κάθε αλφάριθμητο αντικατάστασης προκύπτει για διαφορετική τιμή μετάθεσης του αρχικού αλφαριθμητού, με βάση το κλειδί που τώρα δεν αποτελείται από μια τιμή μετάθεσης αλλά από ένα μοτίβο πολλαπλών τιμών, το οποίο επαναλαμβάνεται. Για παράδειγμα, το «*a*» μπορεί να κρυπτογραφηθεί ως «*η*» στην αρχή του κειμένου, αλλά ως «*o*» όταν συναντάται σε άλλο σημείο του κειμένου. Οι αλγόριθμοι πολυαλφαριθμητικής αντικατάστασης έχουν το πλεονέκτημα ότι κρύβουν τη συχνότητα εμφάνισης των γραμμάτων της γλώσσας που χρησιμοποιήθηκε στο αρχικό κείμενο. Επομένως, ο κρυπταναλυτής δεν μπορεί να χρησιμοποιήσει την πληροφορία της συχνότητας εμφάνισης των γραμμάτων της γλώσσας ώστε να συνάγει ανάλογη πληροφορία για τα γράμματα του κρυπτοκείμενου.

1.2.2 Αντιμετάθεση

Οι αλγόριθμοι αντιμετάθεσης (transposition ή permutation), είναι μια κατηγορία κρυπτογραφικών αλγορίθμων που περιλαμβάνουν την αναδιάταξη των θέσεων των χαρακτήρων ή των στοιχείων στο αρχικό μήνυμα. Αντί να αντικαθίστούν μεμονωμένα γράμματα ή σύμβολα με άλλα, όπως στους αλγορίθμους αντικατάστασης, οι αλγόριθμοι αντιμετάθεσης επικεντρώνονται στην αλλαγή της σειράς με την οποία εμφανίζονται οι χαρακτήρες. Όπως και με τους αλγορίθμους αντικατάστασης, έτσι και αυτοί οι αλγόριθμοι αποτελούν θεμελιώδη έννοια στον τομέα της κλασικής κρυπτογραφίας και χρησιμεύουν ως ένα από τα δομικά στοιχεία για πιο σύνθετες τεχνικές κρυπτογράφησης.

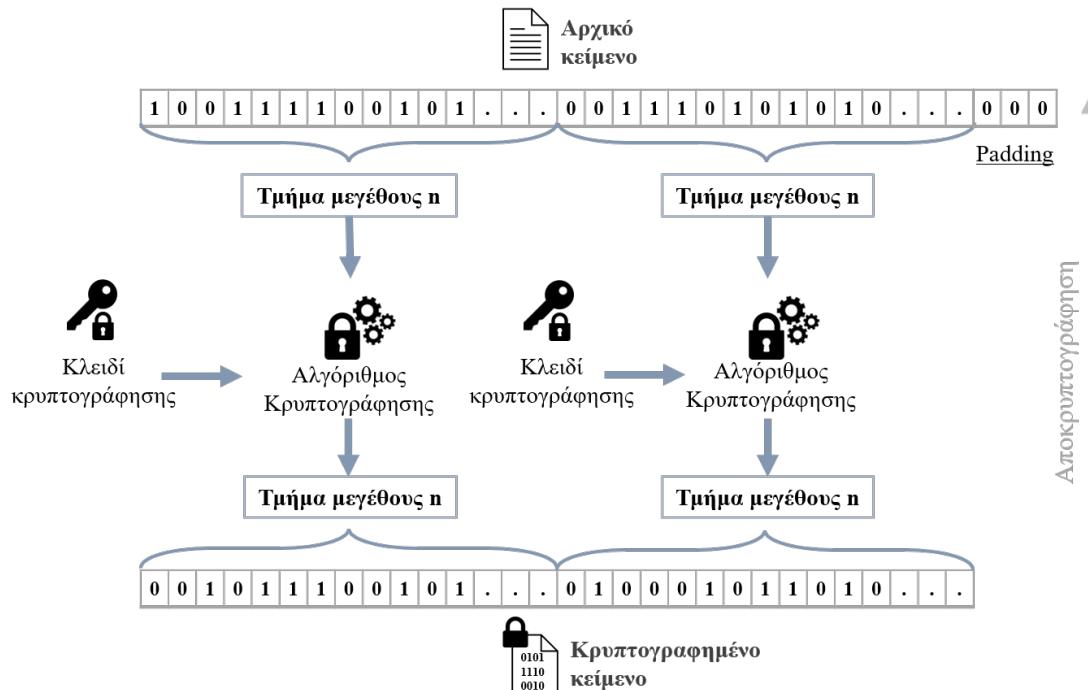
Σε έναν αλγόριθμο αντιμετάθεσης, το κλειδί καθορίζει τους κανόνες για την αναδιάταξη των χαρακτήρων στο αρχικό κείμενο. Πρόκειται για σχετικά απλούς αλγορίθμους που μπορούν να σπάσουν με επίθεση εξαντλητικής έρευνας (brute force attack) εάν ο χώρος του κλειδιού δεν είναι αρκετά μεγάλος.

1.3 Κρυπτογραφικοί Αλγόριθμοι Μπλοκ

Οι κρυπτογραφικοί αλγόριθμοι μπλοκ (block cipher), γνωστοί και ως κρυπταλγόριθμοι μπλοκ ή τμήματος, αποτελούν ένα από τα τρία βασικά αρχέτυπα στη συμμετρική κρυπτογραφία, με τα άλλα δύο να είναι οι κρυπτογραφικοί αλγόριθμοι ροής και οι συναρτήσεις σύνοψης. Παρέχουν μια αντιστρεπτή, με βάση την τιμή του κρυπτογραφικού κλειδιού, αντιστοίχιση ενός τμήματος (ή μπλοκ) δεδομένων αρχικού κειμένου σε ένα ίδιου μεγέθους τμήμα κρυπτοκειμένου. Χρησιμοποιούνται ευρέως για κρυπτογράφηση δεδομένων, εξασφαλίζοντας την εμπιστευτικότητά τους. Αποτελούν επίσης τη βάση για την παραγωγή κωδίκων αυθεντικοποίησης μηνύματος, γνωστών και ως κώδικες επαλήθευσης (tautóτητας) μηνυμάτων (Message Authentication Code – MAC, βλέπε Ενότητα 5.2), που εξασφαλίζουν την ακεραιότητα των δεδομένων, καθώς επίσης και για την επαλήθευση της ταυτότητας της πηγής των μηνυμάτων. Ορισμένες συναρτήσεις σύνοψης δημιουργούνται επίσης από κρυπτογραφικούς αλγορίθμους μπλοκ.

Οι κρυπτογραφικοί αλγόριθμοι μπλοκ αρχικά κομματιάζουν το αρχικό κείμενο σε μπλοκ σταθερού μήκους, το οποίο ορίζεται από τον εκάστοτε αλγόριθμο. Στη συνέχεια, κρυπτογραφούν ένα μπλοκ κάθε φορά χρησιμοποιώντας το ίδιο κλειδί κρυπτογράφησης για καθένα από αυτά (Σχήμα 1.2). Το αποτέλεσμα αυτής

της διαδικασίας είναι ένας αριθμός κρυπτογραφημένων μπλοκ, από την συνένωση των οποίων προκύπτει το κρυπτοκείμενο.



Σχήμα 1.2: Κρυπτογράφηση κατά μπλοκ.

Κατά τη φάση του κομματιάσματος του αρχικού κειμένου, εάν το μήκος του τελευταίου μπλοκ είναι μικρότερο από το απαιτούμενο μήκος μπλοκ, ακολουθείται μια διαδικασία προσθήκης συμπληρωματικών bit(s) που ονομάζεται πλήρωση των δεδομένων (padding). Αν και κάθε αλγόριθμος μπλοκ υποστηρίζει ένα συγκεκριμένο μέγεθος μπλοκ κειμένου, συνήθως υποστηρίζει πολλαπλά μεγέθη κλειδιών. Για παράδειγμα, ο αλγόριθμος Advanced Encryption Standard (AES), ένας τυποποιημένος και ευρέως χρησιμοποιούμενος αλγόριθμος, χρησιμοποιεί μπλοκ μεγέθους 128-bit και υποστηρίζει μεγέθη κλειδιών 128, 192 και 256-bit.

Το μέγεθος μπλοκ είναι μια σημαντική παράμετρος. Κάτω από ορισμένες συνθήκες, και ανάλογα με την ποσότητα των κρυπτογραφημένων δεδομένων, ένα μικρό μέγεθος μπλοκ μπορεί να οδηγήσει σε διαρροή πληροφοριών.

Τα τελευταία χρόνια έχει προκύψει πληθώρα διαθέσιμων αλγορίθμων μπλοκ. Από τον πλέον ανασφαλή Data Encryption Standard (DES) που προτάθηκε στη δεκαετία του '70, περάσαμε πλέον στους Kasumi, Blowfish, Serpent, Camellia, και τον «αντικαταστάτη» του DES, ως αποτέλεσμα σχετικού διαγωνισμού που έγινε από το National Institute of Standards and Technology (NIST), τον Advanced Encryption Standard (AES) [2, 3]. Δυστυχώς, δε θεωρούνται όλοι ασφαλείς, ειδικά για μελλοντική χρήση. Οι αλγόριθμοι που θεωρούνται ασφαλείς για μελλοντική χρήση έχουν μελετηθεί εκτενώς και θεωρούνται ασφαλείς έναντι γνωστών επιθέσεων και κρυπταναλυτικών μεθόδων, συνοδεύονται από κάποια απόδειξη ασφαλείας, και αναμένεται να παραμείνουν ασφαλείς για τα επόμενα 10-50 χρόνια [4].

Παρόλο που δεν υπάρχει απόλυτη συναίνεση, μελέτες του οργανισμού ENISA [5], και του Ευρωπαϊκού έργου ECRYPT [4], προτείνουν τον περιορισμό των επιλογών κρυπτογραφικών αλγορίθμων σε αυτούς που έχουν μελετηθεί επαρκώς. Οι ίδιες μελέτες προτείνουν μόνο τη χρήση των κρυπταλγορίθμων AES, Camellia και Serpent σε νέα συστήματα για τα επόμενα 10-50 χρόνια (μελλοντική χρήση). Στον Πίνακα 1.1 παρουσιάζεται η προτεινόμενη από τις παραπάνω μελέτες χρήση των κρυπτογραφικών αλγορίθμων μπλοκ. Η κατηγοριοποίηση «παλαιού τύπου» (legacy) αφορά αλγορίθμους ή μηχανισμούς που χρησιμοποιούνται ήδη και για τους οποίους, είτε δεν υπάρχει γνωστή αδυναμία ωστόσο υπάρχουν καλύτερες εναλλακτικές (σύμβολο '✓'),

είτε συνιστάται άμεση αντικατάσταση τους (σύμβολο 'X'). Αντιστοίχως, η κατηγοριοποίηση «Μελλοντική χρήση» αφορά τη χρήση του αλγορίθμου ή μηχανισμού σε νέα ή μελλοντικά συστήματα η οποία είτε είναι αποδεκτή για τα επόμενα 10-50 χρόνια γιατί ο αλγόριθμος ή μηχανισμός θεωρείται (αποδεδειγμένα) ασφαλής (σύμβολο '✓'), είτε δε συνιστάται (σύμβολο 'X').

Πίνακας 1.1: Κρυπτογραφικοί αλγόριθμοι μπλοκ.

Αλγόριθμος	Κατηγοριοποίηση Παλαιού Τύπου	Κατηγοριοποίηση Μελλοντική Χρήση	Μέγεθος Τμήματος	Μέγεθος Κλειδιού
AES	✓	✓	128	128, 192, 256
Camellia	✓	✓	128	128, 192, 256
Serpent	✓	✓	128	128, 192, 256
3DES (με τριπλό κλειδί)	✓	X	64	168
2DES (με διπλό κλειδί)	✓	X	64	112
Kasumi	✓	X	64	128
Blowfish \geq 80-bit	✓	X	64	80 - 448
DES	X	X	64	56

Παρά την ύπαρξη καλύτερων εναλλακτικών λύσεων, οργανισμοί με ήδη εγκατεστημένα κρυπτοσυστήματα που βασίζονται σε αλγορίθμους, όπως Blowfish, Kasumi ή 3DES, συνεχίζουν να τα χρησιμοποιούν. Τέτοιου είδους αλγόριθμοι θεωρούνται αρκετά ασφαλείς για χρήση σήμερα και στο εγγύς μέλλον, και η αντικατάστασή τους ενδέχεται να μην αποτελεί εφικτή λύση, κυρίως λόγω διαλειτουργικότητας με άλλα εγκατεστημένα συστήματα και εφαρμογές, ή λόγω οικονομικών ζητημάτων. Παρ' όλα αυτά, συνιστάται η μετάπτωση σε αλγορίθμους οι οποίοι θεωρούνται ασφαλείς για μελλοντική χρήση.

Σημειώστε ότι ο Three-key-3DES είναι ο αλγόριθμος DES που εφαρμόζεται τρεις φορές σε κάθε μπλοκ, χρησιμοποιώντας κάθε φορά ένα διαφορετικό κλειδί και ως εκ τούτου μπορούμε να πούμε ότι χρησιμοποιεί ένα κλειδί 168-bit. Η κρυπτογράφηση ενός μηνύματος m ακολουθεί τη διαδικασία της κρυπτογράφησης, αποκρυπτογράφησης, και εκ νέου κρυπτογράφησης του μηνύματος με τα τρία διαφορετικά κλειδιά K1, K2 και K3 ώστε να προκύψει το κρυπτοκείμενο $c = E_{K3}(D_{K2}(E_{K1}(m)))$. Η διαδικασία αυτή είναι γνωστή και ως πολλαπλή κρυπτογράφηση. Ο DES δύο κλειδιών ακολουθεί μια παρόμοια προσέγγιση αλλά χρησιμοποιεί μόνο δύο κλειδιά, K1 και K2 για να εκτελέσει τη διαδικασία της κρυπτογράφησης ως $c = E_{K1}(D_{K2}(E_{K1}(m)))$. Αυτές οι παραλλαγές εισήχθησαν στο παρελθόν για να επιτρέψουν τη συνέχιση της χρήσης του άλλοτε ισχυρού αλγορίθμου DES, παρά το γεγονός ότι ήταν πλέον ευάλωτος σε επιθέσεις εξαντλητικής αναζήτησης κλειδιών, επίσης γνωστή ως επίθεση brute force, λόγω του πολύ μικρού μεγέθους του κλειδιού (δηλ. 56 bits) και των εξελίξεων στην ταχύτητα επεξεργασίας και αποθήκευσης στα υπολογιστικά συστήματα.

1.3.1 Κώδικας Feistel

Ο κώδικας Feistel (Feistel Cipher), επίσης κοινώς γνωστός ως δίκτυο Feistel (Feistel network) δεν αποτελεί έναν συγκεκριμένο αλγόριθμο κρυπτογράφησης μπλοκ, αλλά ένα μοντέλο σχεδίασης από το οποίο προκύπτουν πολλοί διαφορετικοί αλγόριθμοι κρυπτογράφησης μπλοκ. Πήρε το όνομά του από τον γερμανικής καταγωγής φυσικό και κρυπτογράφο Horst Feistel που έκανε πρωτοποριακή έρευνα ενώ εργαζόταν για την IBM στις ΗΠΑ.

Ένα δίκτυο Feistel είναι μια επαναληπτική διαδικασία κρυπτογράφησης με μια εσωτερική συνάρτηση που

ονομάζεται κυκλική συνάρτηση (round function). Η διαδικασία κρυπτογράφησης χρησιμοποιεί τη δομή Feistel που αποτελείται από πολλές επαναλήψεις επεξεργασίας (rounds) της κυκλικής συνάρτησης πάνω στο αρχικό κείμενο και στα παράγωγα αυτού. Η κυκλική συνάρτηση περιλαμβάνει ένα κώδικα γινομένου (product cipher), δηλαδή έναν συνδυασμό αλγορίθμων αντικατάστασης και αντιμετάθεσης. Ο τρόπος λειτουργίας του Κώδικα Feistel βασίζεται στα ακόλουθα βήματα τα οποία επίσης απεικονίζονται στο Σχήμα 1.3:

- **Επέκταση κλειδιού:** Η διαδικασία κρυπτογράφησης ξεκινά με την επέκταση κλειδιού, όπου το αρχικό μυστικό κλειδί μετατρέπεται σε ένα σύνολο δευτερεύοντων κλειδιών, διαφορετικό για κάθε επανάληψη της κρυπτογράφησης.
- **Κυκλική συνάρτηση:** Σε κάθε επανάληψη, το ήμισυ των δεδομένων υποβάλλεται σε επεξεργασία από μια κυκλική συνάρτηση f που λαμβάνει και τα δεδομένα και το δευτερεύον κλειδί της επανάληψης ως είσοδο. Οι κύριες λειτουργίες αυτής της συνάρτησης περιλαμβάνουν κάποια διαδικασία αντικατάστασης δεδομένων (με τη χρήση των επονομαζόμενων S-boxes) και αντιμετάθεσης αυτών (με τη χρήση P-boxes). Το S-box (Substitution Box) είναι ένας μηχανισμός αντικατάστασης που παίρνει μια είσοδο bit και την αντικαθιστά με μια άλλη ακολουθία bits, με σκοπό την αύξηση της ασφάλειας μέσω της προσθήκης πολυπλοκότητας στις σχέσεις μεταξύ του κλειδιού κρυπτογράφησης και του κρυπτογραφημένου κειμένου. Αντίστοιχα, τα P-boxes (Permutation Boxes) είναι ένας μηχανισμός που χρησιμοποιούνται για την αναδιάταξη στις θέσεις των bits ενός μπλοκ δεδομένων, προσθέτοντας έτσι πολυπλοκότητα και κάνοντας τη σχέση μεταξύ του αρχικού κειμένου και του κρυπτογραφημένου κειμένου πιο ασαφή.

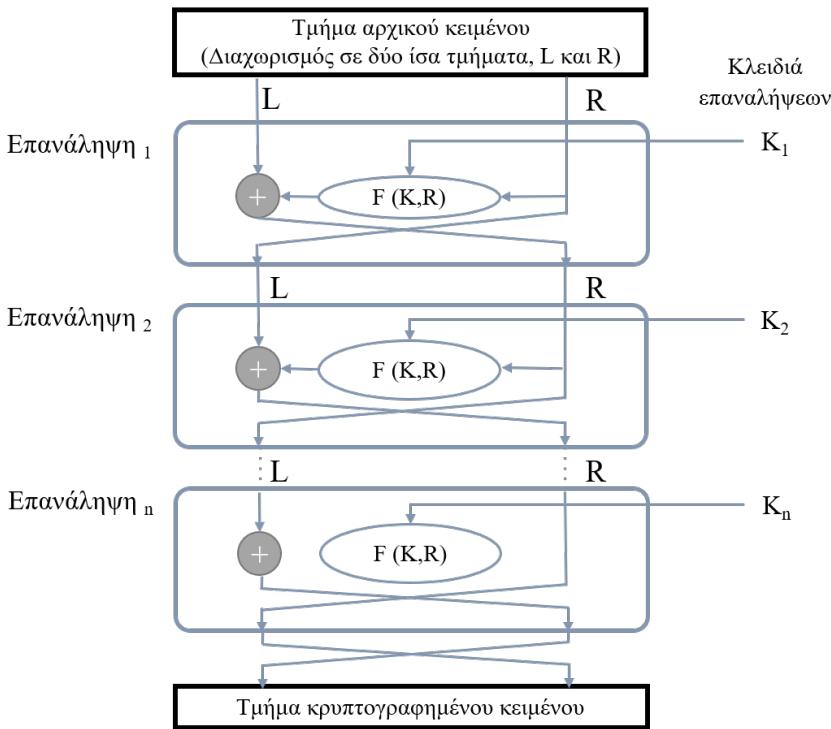
Πιο συγκεκριμένα, το μπλοκ εισόδου σε κάθε επανάληψη χωρίζεται σε δύο τμήματα που μπορούν να συμβολίζονται ως L και R για το αριστερό μισό και το δεξί μισό του μπλοκ, αντίστοιχα. Το R , μαζί με το μυστικό κλειδί K για την δεδομένη επανάληψη, δίνονται ως είσοδος στη συνάρτηση (κρυπτογράφησης) f . Η συνάρτηση παράγει την έξοδο $f(R, K)$ η οποία με τη σειρά της γίνεται είσοδος σε μια πράξη XOR (exclusive-OR) μαζί με το αριστερό τμήμα L . Αυτό διασφαλίζει ότι ακόμη και μικρές αλλαγές στο ένα μισό έχουν σημαντικό αντίκτυπο στο άλλο μισό.

- **Εναλλαγή L και R :** Τα δύο μισά των δεδομένων αμοιβαία εναλλάσσουν θέσεις και η διαδικασία επαναλαμβάνεται για τον επόμενο κύκλο (επανάληψη). Αυτή η εναλλαγή διασφαλίζει ότι οι διαδικασίες κρυπτογράφησης και αποκρυπτογράφησης είναι συμμετρικές.
- **Επαναλήψεις:** Τα παραπάνω βήματα σχηματίζουν μια επανάληψη. Ο αριθμός των επαναλήψεων καθορίζεται από τον εκάστοτε αλγόριθμο κρυπτογράφησης που χρησιμοποιεί τη δομή Feistel.
- **Μόλις ολοκληρωθεί η τελευταία επανάληψη,** τότε τα δύο τμήματα, R και L ενώνονται με αυτή τη σειρά για να σχηματίσουν το μπλοκ κρυπτοκειμένου.

Πληθώρα αλγορίθμων τύπου μπλοκ βασίζονται στον κώδικα Feistel, είτε στη βασική του μορφή, είτε στον ασύμμετρο κώδικα Feistel (Unbalanced Feistel cipher) σύμφωνα με τον οποίο τα τμήματα L_0 και R_0 δεν είναι ίδιουν μεγέθουν. Παραδείγματα αυτών αποτελούν οι αλγόριθμοι: Serpent, Blowfish, CAST-256, DES, Triple DES (3DES), Twofish και MARS.

1.3.2 AES

To 1997, το Εθνικό Ινστιτούτο Προτυποποίησης και Τεχνολογίας των ΗΠΑ (National Institute of Standards and Technology – NIST) ξεκίνησε τη διαδικασία για την ανάπτυξη ενός νέου προτύπου κρυπτογραφικού αλγορίθμου μπλοκ που θα ονομαζόταν Advanced Encryption Standard ή AES. Ο αλγόριθμος AES έπρεπε να λειτουργεί σε μπλοκ των 128-bit και να υποστηρίζει τρία μεγέθη κλειδιών: 128, 192 και 256 bits. Τον Σεπτέμβριο του 1997, το NIST έλαβε 15 προτάσεις. Αφού διοργάνωσε δύο ανοιχτές διασκέψεις για να



Σχήμα 1.3: Ο κώδικας Feistel.

συζητήσουν τις προτάσεις, το 1999 το NIST μείωσε τη λίστα των υποψήφιων προτάσεων σε πέντε.. Ακολούθησε ένας ακόμη γύρος έντονης κρυπτανάλυσης, που κορυφώθηκε στο συνέδριο AES3 τον Απρίλιο του 2000, στο οποίο ένας εκπρόσωπος κάθε μιας από τις ομάδες έκανε μια παρουσίαση υποστηρίζοντας γιατί το πρότυπό τους πρέπει να επιλεγεί ως AES. Τον Οκτώβριο του 2000, το NIST ανακοίνωσε ότι ο Rijndael, ένας κρυπτογραφικός αλγόριθμος από το Βέλγιο, είχε επιλεγεί ως το νέο πρότυπο. Ο AES έγινε επίσημο πρότυπο τον Νοέμβριο του 2001 όταν δημοσιεύθηκε ως πρότυπο του NIST στο FIPS 197 [6]. Αυτό ολοκλήρωσε μια πενταετή διαδικασία για την τυποποίηση του αντικαταστάτη του DES.

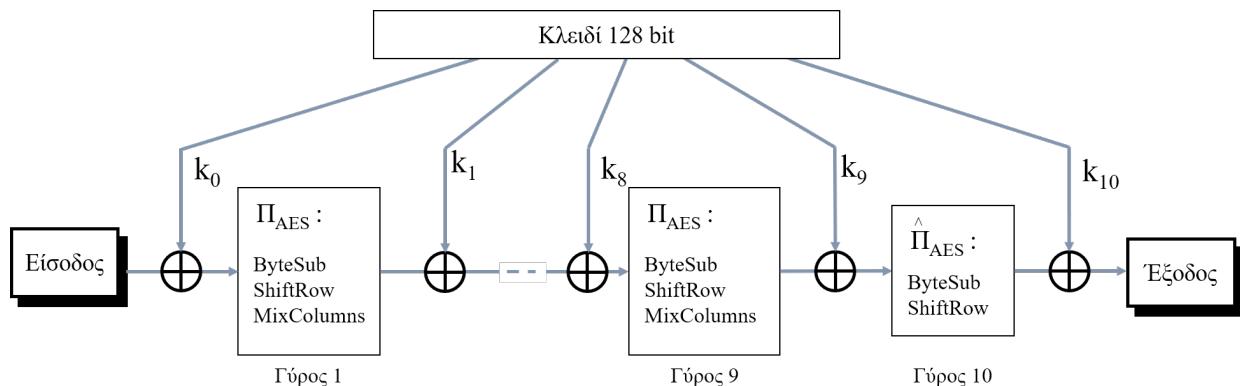
Ο Rijndael σχεδιάστηκε από τους Βέλγους κρυπτογράφους Joan Daemen και Vincent Rijmen [7]. Ο AES είναι ελαφρώς διαφορετικός από τον αρχικό αλγόριθμο Rijndael. Για παράδειγμα, ο Rijndael υποστηρίζει μπλοκ μεγέθους 128, 192 ή 256 bit, ενώ ο AES μόνο μπλοκ μεγέθους 128-bit. Αναφορικά με τα κλειδιά, ο AES υποστηρίζει 3 μεγέθη κλειδιών: 128, 192 και 256 bits (Πίνακας 1.2). Όπως πολλοί αλγόριθμοι κρυπτογράφησης, έτσι και ο AES χρησιμοποιεί μια επαναλαμβανόμενη διαδικασία όπου μια απλή διαδικασία κρυπτογράφησης επαναλαμβάνεται αρκετές φορές. Ο αριθμός των επαναλήψεων εξαρτάται από το μέγεθος του μυστικού κλειδιού. Έτσι για τα μεγέθη κλειδιών 128, 192 και 256 bits απαιτούνται 10, 12 και 14 επαναλήψεις, αντίστοιχα. Ως εκ τούτου, οι εκδόσεις με μεγαλύτερο μήκος κλειδιού είναι πιο αργές (κατά 20% και 40%, αντίστοιχα).

Για παράδειγμα, η δομή του αλγορίθμου AES-128 με τις δέκα επαναλήψεις, φαίνεται στο Σχήμα 1.4. Το Π_{AES} είναι μια σταθερή αντιμετάθεση (permutation), μια συνάρτηση ένα προς ένα στο $\{0, 1\}^{128}$ που δεν εξαρτάται από το κλειδί. Το τελευταίο βήμα κάθε επανάληψης είναι μια πράξη αποκλειστικού-Η (XOR) μεταξύ του κλειδιού του τρέχοντος γύρου με την έξοδο του AES. Αυτό επαναλαμβάνεται 9 φορές ενώ στην τελευταία επανάληψη χρησιμοποιείται ελαφρώς τροποποιημένη αντιμετάθεση του αλγορίθμου AES. Οι αλγόριθμοι που ακολουθούν τη δομή που φαίνεται στο Σχήμα 1.4 είναι γνωστοί ως επαναληπτικοί αλγόριθμοι κρυπτογράφησης Even-Mansour.

Η αντιμετάθεση Π_{AES} αποτελείται από μια ακολουθία τριών αναστρέψιμων πράξεων στο σύνολο $\{0, 1\}^{128}$. Η είσοδος των 128 bits είναι οργανωμένη ως ένας πίνακας κελιών 4×4 , όπου κάθε κελί είναι οκτώ bits. Οι ακόλουθες τρεις αναστρέψιμες λειτουργίες εκτελούνται στη συνέχεια, η μία μετά την άλλη, στον

Πίνακας 1.2: Χαρακτηριστικά του αλγορίθμου AES.

Αλγόριθμος	Μέγεθος Κλειδιού	Μέγεθος Μπλοκ	Αριθμός Επαναλήψεων
AES-128	128	128	10
AES-192	192	128	12
AES-256	256	128	14



Σχήμα 1.4: Ο αλγόριθμος AES.

πίνακα 4×4 .

1. SubBytes. Έστω $S: \{0,1\}^8 \rightarrow \{0,1\}^8$ μια σταθερή αντιμετάθεση, η οποία για την περίπτωση του AES ορίζεται ως ένας hard-coded πίνακας, με 256 entries. Έχει σχεδιαστεί ώστε να μην έχει σταθερά σημεία, δηλαδή $\forall x \in \{0,1\}^8, S(x) \neq x$ καθώς και όχι αντίστροφα σταθερά σημεία, δηλαδή $S(x) \neq \bar{x}$, όπου \bar{x} είναι το bit-wise συμπλήρωμα του x .
2. ShiftRows. Σε αυτό το βήμα εκτελείται μια κυκλική μετατόπιση στις τέσσερις σειρές του πίνακα εισόδου 4×4 : η πρώτη σειρά παραμένει αμετάβλητη, η δεύτερη σειρά μετατοπίζεται κυκλικά κατά ένα byte προς τα αριστερά, η τρίτη σειρά μετατοπίζεται κυκλικά δύο bytes και η τέταρτη σειρά κατά τρία bytes. Σχηματικά, αυτό το βήμα έχει το ακόλουθο αποτέλεσμα:

$$\begin{pmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_4 & \alpha_5 & \alpha_6 & \alpha_7 \\ \alpha_8 & \alpha_9 & \alpha_{10} & \alpha_{11} \\ \alpha_{12} & \alpha_{13} & \alpha_{14} & \alpha_{15} \end{pmatrix} \rightarrow \begin{pmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_5 & \alpha_6 & \alpha_7 & \alpha_4 \\ \alpha_{10} & \alpha_{11} & \alpha_8 & \alpha_9 \\ \alpha_{15} & \alpha_{12} & \alpha_{13} & \alpha_{14} \end{pmatrix}$$

3. MixColumns. Σε αυτό το βήμα, ο πίνακας (array) 4×4 αντιμετωπίζεται ως μήτρα και πολλαπλασιάζεται με έναν σταθερό πίνακα στο πεπερασμένο πεδίο $GF(2^8)$. Στοιχεία στο πεδίο $GF(2^8)$ αναπαρίστανται ως πολυώνυμα στο $GF(2^8)$ βαθμού μικρότερον από οκτώ, όπου ο πολλαπλασιασμός γίνεται modulo το πολυώνυμο $x^8 + x^4 + x^3 + x + 1$.

Η αντιμετάθεση Π_{AES} που χρησιμοποιεί ο AES και απεικονίζεται στο Σχήμα 1.4 είναι η διαδοχική σύνθεση των τριών αντιμεταθέσεων, SubBytes, ShiftRows και MixColumns με αυτή τη σειρά. Στον τελευταίο γύρο ο AES χρησιμοποιεί μια ελαφρώς διαφορετική συνάρτηση που ονομάζουμε $\hat{\Pi}_{AES}$. Αυτή η συνάρτηση είναι η ίδια με το Π_{AES} εκτός από το ότι παραλείπεται το βήμα MixColumns. Αυτή η παράλειψη γίνεται έτσι ώστε η διαδικασία αποκρυπτογράφησης του AES να μοιάζει κάπως με τη διαδικασία κρυπτογράφησης.

1.3.3 Camellia

Ο κρυπτογραφικός αλγόριθμος Camellia αναπτύχθηκε από κοινού από τη Nippon Telegraph and Telephone Corporation και τη Mitsubishi Electric Corporation το 2000. Χρησιμοποιεί μπλοκ μεγέθους 128 bits και υποστηρίζει 3 μήκη κλειδιών: 128, 192 και 256 bits [8]. Οι εκδόσεις με κλειδί 192 ή 256 bit είναι 33% πιο αργές από τις εκδόσεις με κλειδί 128 bits. Ο Camellia χρησιμοποιείται ως μία από τις πιθανές σούίτες κρυπτογράφησης στο TLS και σε αντίθεση με τον AES βασίζεται στον κώδικα Feistel.

Ο Camellia έχει σχεδιαστεί για να επιτρέπει ευελιξία σε εφαρμογές λογισμικού και υλικού σε επεξεργαστές των 32-bit και σε πολλές εφαρμογές, σε επεξεργαστές των 8-bit που χρησιμοποιούνται σε έξυπνες κάρτες (smart cards), κρυπτογραφικό υλισμικό (hardware), και ενσωματωμένα συστήματα.

1.3.4 Serpent

Ο Serpent [9] ήταν ένας από τους υποψήφιους φιναλίστ του AES. Είναι ένας από τους αλγορίθμους που έχουν τυποποιηθεί για το SSH [10]. Ο αλγόριθμος κρυπτογράφησης ροής SOSEMANUK επαναχρησιμοποιεί τμήματα του Serpent στο σχεδιασμό του.

Ο Serpent χρησιμοποιεί τμήματα μεγέθους 128 bits και υποστηρίζει μήκη κλειδιού 128, 192 και 256 bits. Η καλύτερη επίθεση εναντίον του Serpent είναι μια βασική επίθεση ανάκτησης κλειδιού που μπορεί να φτάσει τους 11 από τους 32 γύρους [11]. Ο μεγάλος αριθμός γύρων σημαίνει επίσης ότι η απόδοση του λογισμικού είναι σημαντικά πιο αργή από αυτήν του AES.

1.4 Τρόποι Λειτουργίας Αλγορίθμων Μπλοκ

Οι κρυπτογραφικοί αλγόριθμοι μπλοκ έχουν σχεδιαστεί ώστε να μπορούν να κρυπτογραφούν ένα μπλοκ δεδομένων συγκεκριμένου μεγέθους. Στην πράξη όμως, η ποσότητα των δεδομένων που θέλουμε να κρυπτογραφήσουμε είναι συχνά πολύ μεγαλύτερη από το μέγεθος του μπλοκ. Έτσι, θα πρέπει να βρούμε έναν τρόπο να χρησιμοποιήσουμε τον κρυπτογραφικό αλγόριθμο ώστε να κρυπτογραφήσουμε μεγαλύτερες ποσότητες δεδομένων. Αυτή η μέθοδος ονομάζεται τρόπος λειτουργίας (operation mode). Υπάρχουν αρκετοί τρόποι λειτουργίας που μπορούν να μετατρέψουν ένα αλγόριθμο μπλοκ σε ένα αποτελεσματικό κρυπτοσύστημα το οποίο θα μπορεί να κρυπτογραφήσει αρχικά κείμενα οποιουνδήποτε μεγέθους, όπως είναι οι ECB (Electronic Codebook), CBC (Cipher Block Chaining), CTR (Counter), OFB (Output Feedback) και CFB (Cipher Block Chaining). Ο κρυπτογραφικός αλγόριθμος και ο τρόπος λειτουργίας είναι συμπληρωματικές έννοιες και μπορούν να επιλεγούν ξεχωριστά. Με άλλα λόγια, η χρήση κάποιου συγκεκριμένου αλγορίθμου δεν υπαγορεύει τη χρήση συγκεκριμένου τρόπου λειτουργίας.

Με εξαίρεση τον ECB, ο οποίος θεωρείται ότι είναι ασφαλής μόνο για μικρά μηνύματα, αυτοί οι τρόποι λειτουργίας συνιστώνται να χρησιμοποιούνται για υπάρχοντα συστήματα, αλλά όχι για μελλοντική χρήση (Πίνακας 1.3) [4]. Για να παρέχουμε ασφάλεια σε νέα συστήματα και μελλοντική χρήση, υπάρχουν και άλλοι τρόποι λειτουργίας, όπως ο ECB-mask-ECB (EME) ή ο Format Preserving Encryption (FFX).

1.4.1 ECB

Ο τρόπος λειτουργίας Ηλεκτρονικού Βιβλίου Κωδικών (Electronic Codebook – ECB) είναι ο απλούστερος τρόπος λειτουργίας. Δεν προσθέτει καμία πολυπλοκότητα στο κρυπτοσύστημα καθώς απαιτεί απλώς να κρυπτογραφείται κάθε μπλοκ του αρχικού κειμένου ανεξάρτητα το ένα από το άλλο. Η ακολουθία των μπλοκ κρυπτογραφημένου κειμένου σχηματίζει το κρυπτοκείμενο (Σχήμα 1.5α).

Η κρυπτογράφηση $E_K(m)$ του μηνύματος m , με τη χρήση του αλγορίθμου κρυπτογράφησης E και του κλειδιού K, με τον τρόπο λειτουργίας ECB παρουσιάζεται στον Αλγόριθμο 1.1.

Αντιστοίχως, η αποκρυπτογράφηση $D_K(m)$ του μηνύματος m , με τη χρήση του αλγορίθμου αποκρυπτογράφησης D και του κλειδιού K, με τον τρόπο λειτουργίας ECB παρουσιάζεται στον Αλγόριθμο 1.2.

Πίνακας 1.3: Τρόποι λειτουργίας αλγορίθμων μπλοκ.

Σχήμα	Παλαιού Τύπου	Μελλοντική Χρήση	Σημειώσεις
OFB	✓	✗	Δεν απαιτεί πλήρωση δεδομένων
CFB	✓	✗	Δεν απαιτεί πλήρωση δεδομένων
CTR	✓	✗	Δεν απαιτεί πλήρωση δεδομένων
CBC	✓	✗	
ECB	✗	✗	
XTS	✓	✗	
EME	✓	✓	
FFX	✓	✓	

Αλγόριθμος 1.1: Τρόπος λειτουργίας ECB – Κρυπτογράφηση.

Είσοδος: k -bit key K ;
 n -bit μπλοκ αρχικού κειμένου m_1, m_2, \dots, m_t ;
Έξοδος: n -bit μπλοκ κρυπτοκειμένου c_1, c_2, \dots, c_t ;

```

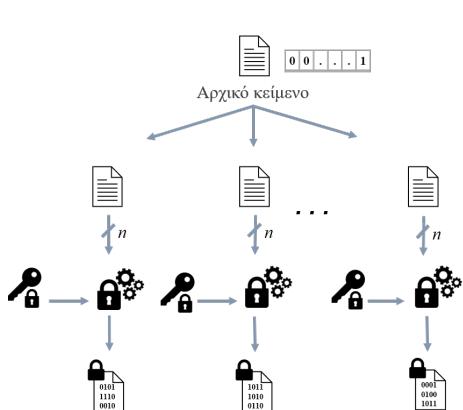
1 for  $1 \leq i \leq t$  do
2   |    $c_i \leftarrow E_K(m_i)$ ;
3 end
4 return  $c_1, c_2, \dots, c_t$ 
```

Αλγόριθμος 1.2: Τρόπος λειτουργίας ECB – Αποκρυπτογράφηση.

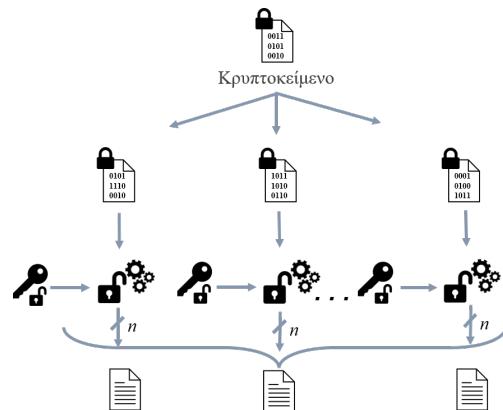
Είσοδος: k -bit key K ;
 n -bit μπλοκ κρυπτοκειμένου c_1, c_2, \dots, c_t ;
Έξοδος: n -bit μπλοκ αρχικού κειμένου m_1, m_2, \dots, m_t ;

```

1 for  $1 \leq i \leq t$  do
2   |    $m_i \leftarrow D_K(c_i)$ ;
3 end
4 return  $m_1, m_2, \dots, m_t$ 
```



(α) Κρυπτογράφηση με ECB.



(β) Αποκρυπτογράφηση με ECB.

Σχήμα 1.5: Τρόπος λειτουργίας ECB.

Ομοίως με τη διαδικασία κρυπτογράφησης, η αποκρυπτογράφηση απαιτεί κάθε τμήμα του κρυπτογραφημένου να αποκρυπτογραφείται ανεξάρτητα (Σχήμα 1.5β).

Ο τρόπος λειτουργίας ECB δε θεωρείται ασφαλής για την κρυπτογράφηση μεγάλου όγκου δεδομένων. Η κρυπτογράφηση κάθε μπλοκ γίνεται ανεξάρτητα από τα υπόλοιπα μπλοκ και επομένως δεν τα επηρεάζει. Αυτό καθιστά τον ECB ευάλωτο σε επιθέσεις αναδιάταξης μπλοκ όπου ο επιτιθέμενος αλλάζει τη σειρά των κρυπτογραφημένων μπλοκ. Για παράδειγμα, ο επιτιθέμενος μπορεί να παρέμβει στο κρυπτογραφημένο μήνυμα το οποίο αποτελείται από τα $c_1, c_2, c_3, \dots, c_t$, και να αλλάξει τη σειρά των c_2 και c_3 ώστε το κρυπτογραφημένο μήνυμα να αποτελείται πλέον από την ακολουθία $c_1, c_3, c_2, \dots, c_t$. Καθώς τα τμήματα κρυπτογραφούνται ανεξάρτητα, οποιαδήποτε αναδιάταξη των μπλοκ κρυπτοκειμένου απλώς οδηγεί κατά την αποκρυπτογράφηση, σε αντίστοιχη αναδιάταξη αρχικού κειμένου. Ανάλογα με το αρχικό μήνυμα, αυτή η αποκρυπτογράφηση ενδέχεται να εξακολουθεί να παρέχει μια αποδεκτή έξοδο. Αξίζει να σημειωθεί ωστόσο ότι το συγκεκριμένο αποτέλει ένα πρόβλημα ακεραιότητας, καθώς γίνεται τροποποίηση του κειμένου, και δεν επηρεάζει την εμπιστευτικότητα του αρχικού κειμένου. Έτσι, μπορεί να αντιμετωπιστεί με τη συνδυαστική χρήση ενός μηχανισμού ακεραιότητας μηνυμάτων.

Ωστόσο, ένα άλλο σημαντικό ζήτημα ασφάλειας του ECB είναι ότι ίδια τμήματα αρχικού κειμένου οδηγούν σε πανομοιότυπα τμήματα κρυπτοκειμένου, καθώς ή κρυπτογράφηση όλων των μπλοκ του αρχικού κειμένου γίνεται με το ίδιο κλειδί κρυπτογράφησης. Αυτό μπορεί να αποκαλύψει μερικές χρήσιμες πληροφορίες σε έναν επιτιθέμενο. Για παράδειγμα, αν ένας επιτιθέμενος γνωρίζει το περιεχόμενο ενός κρυπτογραφημένου μπλοκ (από προηγούμενη γνώση ή αναλυτική επίθεση), τότε μπορεί να αναγνωρίσει τα ακριβή περιεχόμενα όλων των πανομοιότυπων μπλοκ στο κρυπτογραφημένο μήνυμα. Επιπλέον, μπορεί να εντοπίσει επαναλαμβανόμενα τμήματα αρχικού κειμένου (patterns), καθώς αυτά θα αποτελούν επαναλαμβανόμενα μπλοκ κρυπτοκειμένου (ένα παράδειγμα απεικονίζεται στο Σχήμα 1.7. Αυτό δημιουργεί την δυνατότητα για τον επιτιθέμενο να αποκτήσει πληροφορίες για το αρχικό μήνυμα, ακόμα και χωρίς να αποκρυπτογραφήσει ολόκληρο το μήνυμα.

Ως εκ τούτου, αυτή η λειτουργία χρησιμοποιείται σπάνια και πρέπει να αποφεύγεται για μηνύματα των οποίων το μήκος υπερβαίνει το μέγεθος ενός μπλοκ.

1.4.2 CBC

Στον τρόπο λειτουργίας Αλυσιδωτού Μπλοκ (Cipher Block Chaining – CBC), κάθε τμήμα αρχικού κειμένου πριν την κρυπτογράφησή του γίνεται είσοδος σε μια πράξη XOR μαζί με το προηγούμενο μπλοκ κρυπτοκειμένου. Το αποτέλεσμα της πράξης κρυπτογραφείται χρησιμοποιώντας τον επιλεγμένο αλγόριθμο κρυπτογράφησης τμήματος (Αλγόριθμος 1.3), όπως απεικονίζεται στο Σχήμα 1.6α. Για το πρώτο μπλοκ, η πράξη XOR γίνεται με ένα μπλοκ που ονομάζεται Διάνυσμα Αρχικοποίησης (Initialisation Vector – IV). Το IV δε χρειάζεται να είναι μυστικό, αλλά πρέπει να είναι απρόβλεπτο.

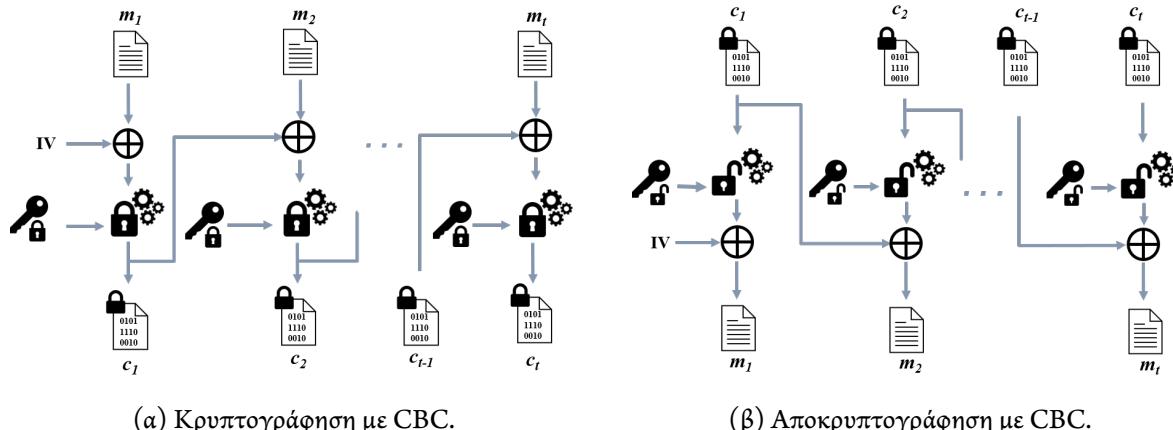
Αλγόριθμος 1.3: Τρόπος λειτουργίας CBC – Κρυπτογράφηση.

Είσοδος: k -bit key K ;
 n -bit διάνυσμα αρχικοποίησης IV;
 n -bit μπλοκ τμήματα αρχικού κειμένου m_1, m_2, \dots, m_t ;

Έξοδος: n -bit μπλοκ κρυπτοκειμένου c_1, c_2, \dots, c_t ;

- 1 $c_1 \leftarrow E_K(IV \oplus m_1);$
- 2 | $c_i \leftarrow E_K(c_{i-1} \oplus m_i);$
- 3 **end**
- 4 **return** c_1, c_2, \dots, c_t

Ο τρόπος λειτουργίας CBC προσφέρει πολλά πλεονεκτήματα έναντι της ECB. Το πιο αξιοσημείωτο είναι ότι στην περίπτωση του CBC, δύο πανομοιότυπα μπλοκ αρχικού κειμένου δεν δίνουν τα ίδια μπλοκ κρυπτοκειμένου και επομένως, δεν αποκαλύπτουν πληροφορίες σχετικά με ο αρχικό κείμενο. Επιπλέον, ενώ η ανα-



Σχήμα 1.6: Τρόπος λειτουργίας CBC.

διάταξη των μπλοκ κρυπτοκειμένου στην λειτουργία ECB μπορεί να οδηγήσει σε καταληπτό ή επεξεργάσιμο αποκρυπτογραφημένο μήνυμα, η αναδιάταξη των μπλοκ που έχουν κρυπτογραφηθεί με CBC θα επηρεάσει τη διαδικασία αποκρυπτογράφησης με αποτέλεσμα μη καταληπτά ή επεξεργάσιμα.

Για την ανάκτηση των τμημάτων αρχικού κειμένου, το μπλοκ κρυπτοκειμένου αποκρυπτογραφείται με την εφαρμογή του αλγορίθμου αποκρυπτογράφησης (Αλγόριθμος 1.4) και αυτό που προκύπτει χρησιμοποιείται ως είσοδος στην πράξη XOR με το προηγούμενο κρυπτογραφημένο τμήμα, όπως απεικονίζεται στο Σχήμα 1.6β. Στη διαδικασία αυτή, εξαίρεση αποτελεί το πρώτο κρυπτογραφημένο τμήμα το οποίο, όπως και στην περίπτωση της κρυπτογράφησης, συνδυάζεται με το διάνυσμα αρχικοποίησης IV.

Αλγόριθμος 1.4: Τρόπος λειτουργίας CBC – Αποκρυπτογράφηση.

Είσοδος: k -bit key K ;

n -bit διάνυσμα αρχικοποίησης IV ;

n -bit μπλοκ κρυπτοκειμένου c_1, c_2, \dots, c_t ;

Έξοδος: n -bit τμήματα αρχικού κειμένου m_1, m_2, \dots, m_t ;

1 $m_1 \leftarrow IV \oplus D_K(c_1)$;

for $2 \leq i \leq t$ do

2 | $m_i \leftarrow c_{i-1} \oplus D_K(c_i)$;

3 end

4 return m_1, m_2, \dots, m_t

Στον τρόπο λειτουργίας CBC, σε αντίθεση με τον ECB, πανομοιότυπα τμήματα αρχικού κειμένου δίνουν διαφορετικά κρυπτογραφημένα μπλοκ, αρκεί τα προηγούμενα σε αυτά κρυπτογραφημένα μπλοκ να είναι διαφορετικά. Στο παράδειγμα που απεικονίζεται στο Σχήμα 1.7, μπορούμε να δούμε ότι η κρυπτογράφηση σε λειτουργία ECB οδηγεί σε ένα κρυπτοκειμένο που μπορεί να αποκαλύψει πληροφορίες σε μη εξουσιοδοτημένες οντότητες σχετικά με το αρχικό κείμενο. Στον CBC, ίδια τμήματα αρχικού κειμένου δίνουν ίδια κρυπτογραφημένα μπλοκ, μόνο όταν έχουν κρυπτογραφηθεί χρησιμοποιώντας το ίδιο κλειδί και το ίδιο διάνυσμα αρχικοποίησης IV. Άλλαζοντας το κλειδί, το IV, ή το πρώτο μπλοκ αρχικού κειμένου παίρνουμε διαφορετικό κρυπτογράφημα.

Η χρήση του τρόπου λειτουργίας CBC έχει ως αποτέλεσμα τα κρυπτογραφημένα μπλοκ c_j να εξαρτώνται από το αρχικό μπλοκ m_j και, λόγω της χρήσης της πράξης XOR, από όλα τα προηγούμενα μπλοκ. Εάν αλλαχτεί η σειρά των κρυπτογραφημένων μπλοκ, θα επηρεαστεί και η αποκρυπτογράφηση.

Διάδοση λάθους: ένα λάθος bit κατά τη μετάδοση του μπλοκ c_j θα επηρεάσει την αποκρυπτογράφηση των τμημάτων c_j και c_{j+1} (αφού το m_j εξαρτάται από τα c_j και c_{j-1}).



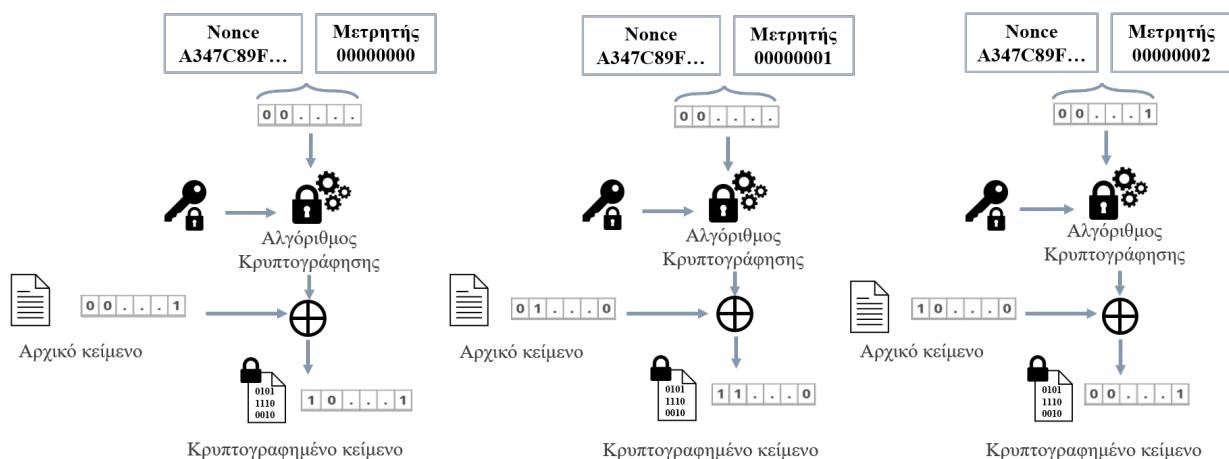
Σχήμα 1.7: Κρυπτογράφηση με τρόπο λειτουργίας ECB και CBC (Πηγή: [CrypTool](#)).

Ανάκτηση λάθους: ο τρόπος λειτουργίας CBC είναι self-synchronizing υπό την έννοια ότι εάν έχουμε ένα λάθος στο τμήμα c_j αλλά όχι στο c_{j+1} , τότε θα έχουμε σωστή αποκρυπτογράφηση του c_{j+2} σε m_{j+1} .

1.4.3 CTR

Στον τρόπο λειτουργίας Μετρητή (Counter – CTR), ένα μπλοκ κρυπτογραφημένου κειμένου προκύπτει ως έξοδος πράξης XOR με εισόδους το μπλοκ αρχικού κειμένου και μια κλειδοροή μεταξύ του τμήματος αρχικού κειμένου και μιας κλειδοροής (keystream), η οποία δημιουργείται κρυπτογραφώντας μια τυχαία ή μη επαναλαμβανόμενη τιμή, γνωστή ως nonce, σε συνδυασμό με έναν μετρητή (Σχήμα 1.8). Σε κάθε επόμενο μπλοκ αρχικού κειμένου, ο μετρητής αυξάνεται. Η τιμή nonce ωστόσο, είναι η ίδια για όλα τα τμήματα ενός συγκεκριμένου μηνύματος.

Η λειτουργία CTR μετατρέπει έναν αλγόριθμο κρυπτογράφησης τμήματος σε αλγόριθμο κρυπτογράφησης ροής. Ως αποτέλεσμα, το τμήμα αρχικού κειμένου δεν χρειάζεται να έχει το ίδιο μήκος με το μέγεθος τμήματος του επιλεγμένου αλγορίθμου. Δεν απαιτείται πλήρωση των δεδομένων (padding). Εάν το μέγεθος ενός τμήματος είναι μικρότερο από το μέγεθος τμήματος του αλγορίθμου, το keystream απλώς περικόπτεται στο μέγεθος τμήματος αρχικού κειμένου.

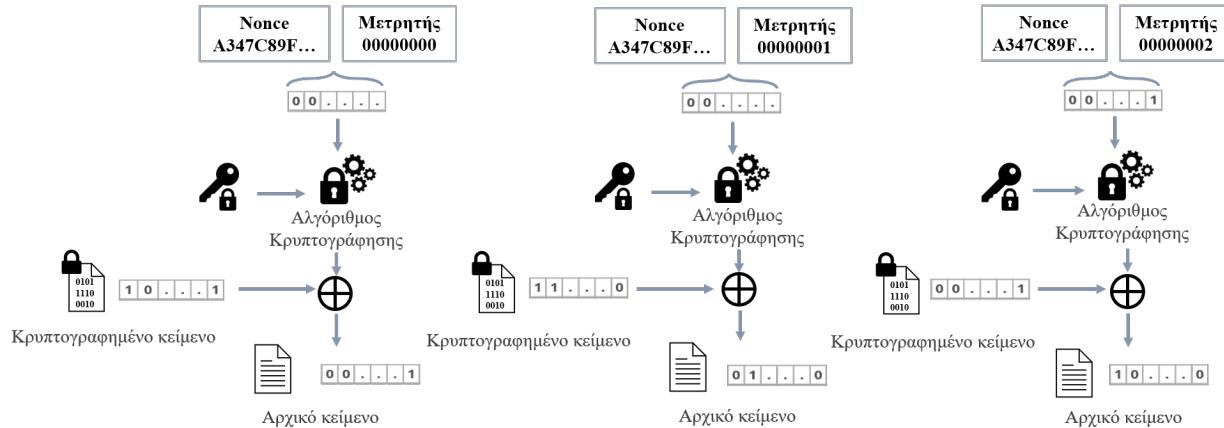


Σχήμα 1.8: Κρυπτογράφηση με τρόπο λειτουργίας CTR.

Η λειτουργία CTR προσφέρει μερικά αξιοσημείωτα πλεονεκτήματα, όπως το ότι οι διαδικασίες κρυπτογράφησης και αποκρυπτογράφησης μπορούν να εκτελούνται παράλληλα. Αυτό επιτρέπει αποτελεσματικές εφαρμογές που επιταχύνουν και τις δύο διαδικασίες. Ένα άλλο πλεονέκτημα είναι η λεγόμενη ιδιότητα τυ-

χαίας πρόσβασης, όπου μπορεί κανείς να αποκρυπτογραφήσει ένα συγκεκριμένο τμήμα από το κρυπτοκείμενο χωρίς να χρειάζεται να αποκρυπτογραφήσει τα προηγούμενα.

Η διαδικασία αποκρυπτογράφησης σε λειτουργία CTR είναι παρόμοια με αυτή της κρυπτογράφησης, και είναι το αποτέλεσμα της πράξης XOR με εισόδους την ίδια κλειδοροή και το κρυπτογραφημένου μπλοκ (Σχήμα 1.9).



Σχήμα 1.9: Αποκρυπτογράφηση με τρόπο λειτουργίας CTR.

1.4.4 CFB

Στον τρόπο λειτουργίας Ανατροφοδότησης Κρυπταλγορίθμου (Cipher-Feedback – CFB), το αρχικό μήνυμα χρησιμοποιείται με τρόπο που επηρεάζει τη διαδικασία κρυπτογράφησης. Η κρυπτογράφηση κάθε μπλοκ αρχικού κειμένου είναι το αποτέλεσμα μιας πράξης XOR με εισόδους το μπλοκ αυτό και ενός τμήματος μιας κλειδοροής (keystream) (Σχήμα 1.10). Η κλειδοροή δημιουργείται επανακρυπτογραφώντας το προηγούμενο κρυπτογραφημένο μπλοκ, με εξαίρεση την κρυπτογράφηση του πρώτου μπλοκ αρχικού κειμένου όπου εκεί χρησιμοποιείται ένα διάνυσμα αρχικοποίησης, όπως αποτυπώνεται και στο Σχήμα 1.10. Από αυτήν την άποψη, η λειτουργία CFB, όπως και η CTR, μπορεί να μετατρέψει έναν αλγόριθμο κρυπτογράφησης τμήματος σε αλγόριθμο ροής.

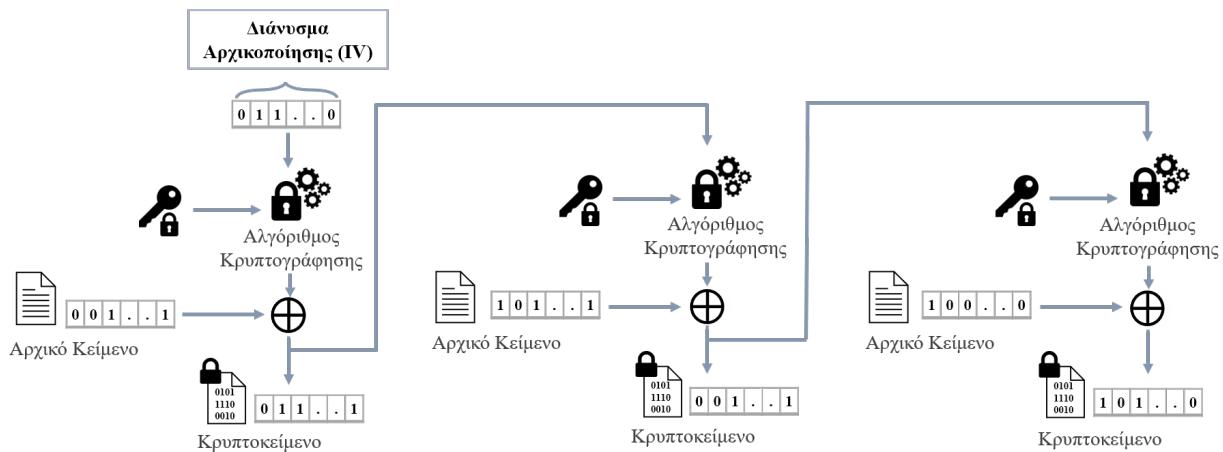
Στο CFB το μέγεθος των τμημάτων αρχικού κειμένου μπορεί να είναι μικρότερο από το μέγιστο επιτρεπόμενο μέγεθος τμήματος των n -bits. Σε μια τέτοια περίπτωση, η διαδικασία κρυπτογράφησης θα χρησιμοποιήσει μόνο τα k πιο αριστερά bits του τμήματος της κλειδοροής, ως είσοδο στην πράξη XOR.

Η αποκρυπτογράφηση CFB είναι πανομοιότυπη με τη διαδικασία κρυπτογράφησης, εκτός από το ότι η πράξη XOR εφαρμόζεται στην κλειδοροή και στο κρυπτογραφημένο μπλοκ (Σχήμα 1.11). Όπως και στο CTR, στη διαδικασία αποκρυπτογράφησης το τμήμα κλειδοροής δημιουργείται με τον ίδιο ακριβώς τρόπο όπως στη διαδικασία κρυπτογράφησης.

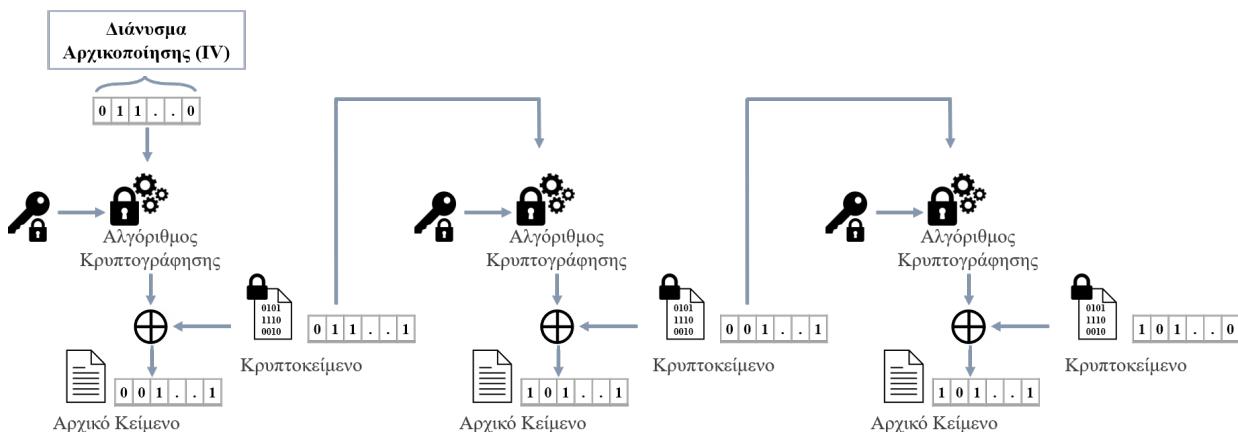
Μία βασική ιδιότητα του CFB είναι ότι τμήματα ενός μηνύματος μπορούν να αποκρυπτογραφηθούν χωρίς να χρειάζεται να ξεκινήσει την αποκρυπτογράφηση από την αρχή του μηνύματος ή να είναι γνωστή η ακριβής θέση του κρυπτογραφημένου τμήματος στη συνολική ροή. Έτσι, οι διαδικασίες κρυπτογράφησης και αποκρυπτογράφησης διαφορετικών μπλοκ μπορούν να τρέχουν παράλληλα.

1.4.5 OFB

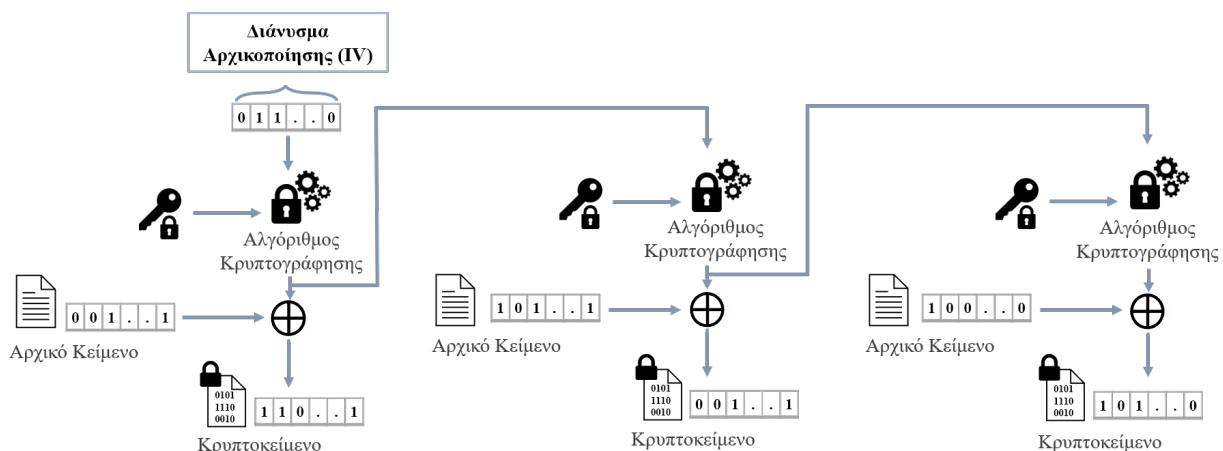
Η λειτουργία ανάδρασης εξόδου, (Output Feedback Mode – OFB), είναι παρόμοια με τη λειτουργία CFB. Η διαφορά με τον τρόπο λειτουργία CFB είναι ότι στο OFB κάθε τμήμα κλειδοροής δημιουργείται με κρυπτογράφηση του προηγούμενου τμήματος της κλειδοροής (Σχήμα 1.12). Το πρώτο τμήμα της κλειδοροής είναι το αποτέλεσμα της κρυπτογράφησης του διανύσματος αρχικοποίησης.



Σχήμα 1.10: Κρυπτογράφηση με τρόπο λειτουργίας CFB.

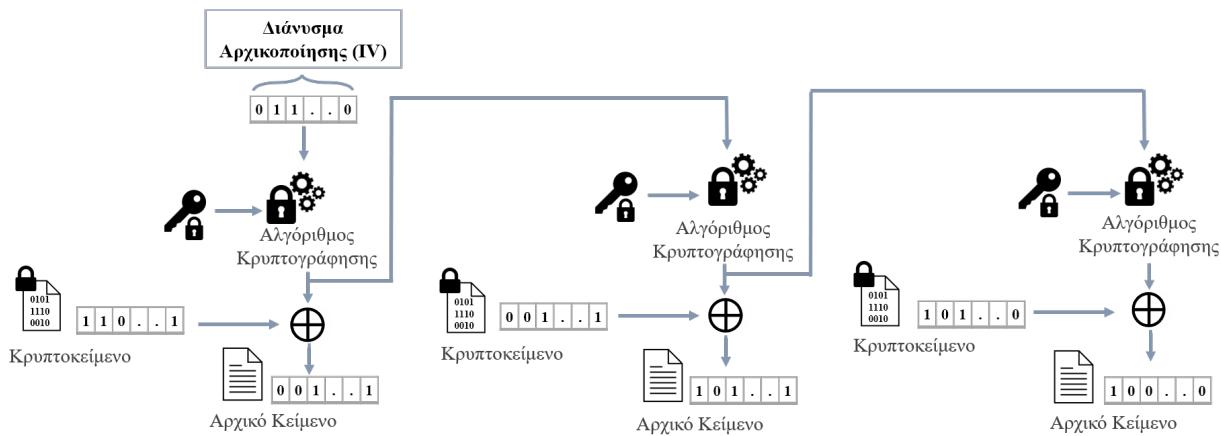


Σχήμα 1.11: Αποκρυπτογράφηση με τρόπο λειτουργίας CFB.



Σχήμα 1.12: Κρυπτογράφηση με τρόπο λειτουργίας OFB.

Παρόμοια με την CFB, η αποκρυπτογράφηση OFB είναι ίδια με τη διαδικασία κρυπτογράφησης, εκτός από το ότι η πράξη XOR εφαρμόζεται στην κλειδοροή και στο κρυπτογραφημένο μπλοκ (Σχήμα 1.13).

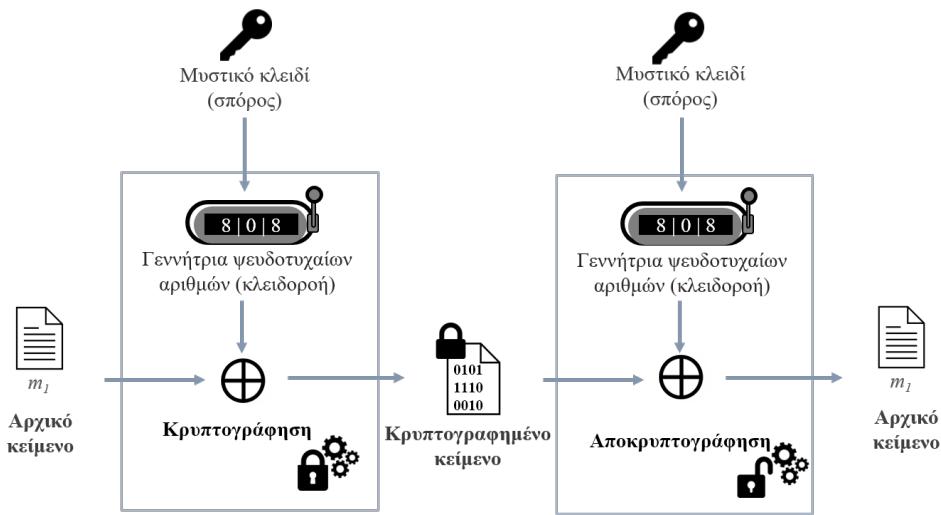


Σχήμα 1.13: Αποκρυπτογράφηση με τρόπο λειτουργίας OFB.

1.5 Αλγόριθμοι Κρυπτογράφησης Ροής

Σε αντίθεση με τους αλγορίθμους κρυπτογράφησης τμήματος, οι αλγόριθμοι κρυπτογράφησης ροής (stream cipher) εκτελούν λειτουργίες κρυπτογράφησης/αποκρυπτογράφησης σε (ακόμη και σε πραγματικού χρόνου) ροές bit. Κρυπτογραφούν το αρχικό κείμενο, το οποίο θεωρείται ως μια ροή bits, εφαρμόζοντας μια πράξη XOR μεταξύ των bits του αρχικού κειμένου και της κλειδοροής (Σχήμα 1.14). Η κλειδοροή συνήθως δημιουργείται από μια γεννήτρια ψευδοτυχαίων αριθμών (Pseudo-Random Number Generator – PRNG), η οποία δημιουργεί μια ακολουθία bit που προσεγγίζει τις ιδιότητες μιας ακολουθίας τυχαίων bit. Οι ιδιότητες τυχαιότητας της κλειδοροής καθορίζουν και την ασφάλεια του κρυπτοσυστήματος.

Οι αλγόριθμοι κρυπτογράφησης ροής είναι γενικά πολύ αποτελεσματικοί και κατάλληλοι για εφαρμογές υλικού και χρησιμοποιούνται για την παροχή εμπιστευτικότητας σε επικοινωνίες υψηλής ταχύτητας, όπως οι τηλεπικοινωνίες.



Σχήμα 1.14: Αλγόριθμος κρυπτογράφησης ροής.

Η γεννήτρια ψευδοτυχαίων αριθμών πρέπει να είναι συγχρονισμένη μεταξύ του αποστολέα και του παραλήπτη της ροής bit. Για αυτό, ελέγχεται από ένα μυστικό κλειδί που μοιράζεται μεταξύ εξουσιοδοτημένων συμμετεχόντων και παρέχεται ως είσοδος στη γεννήτρια ροής κλειδιών. Αυτό το κλειδί είναι επίσης γνωστό ως σπόρος (seed). Χρησιμοποιώντας την ίδια γεννήτρια ψευδοτυχαίων αριθμών και τον ίδιο σπόρο, τα εξου-

σιοδοτημένα μέρη μπορούν να δημιουργήσουν την ίδια ροή κλειδιών. Μια ασφαλής γεννήτρια ψευδοτυχαίων αριθμών θα πρέπει να μπορεί να δημιουργεί μεγάλες περιόδους μη επαναλαμβανόμενων και στατιστικά απρόβλεπτων μοτίβων. Οι γεννήτριες ψευδοτυχαίων αριθμών παρουσιάζονται αναλυτικότερα στο Κεφάλαιο 4.

Ο αλγόριθμος κρυπτογράφησης Vernam, ο οποίος προτάθηκε στα μέσα της δεκαετίας του 1910, ονομάζεται επίσης ως one-time-pad είναι ένας αλγόριθμος κρυπτογράφησης ροής όπου το μήκος του κλειδιού είναι τουλάχιστον ίσο με το μήκος του μηνύματος. Εάν, πέραν του μήκους κλειδιού, τα ψηφία του κλειδιού είναι τυχαία και ανεξάρτητα το ένα από το άλλο, ο αλγόριθμος λέγεται ότι παρέχει τέλεια μυστικότητα (perfect secrecy). Ένας αλγόριθμος κρυπτογράφησης προσφέρει τέλεια μυστικότητα εάν το αρχικό κείμενο και το κρυπτοκείμενο είναι στατιστικά ανεξάρτητα, δηλαδή το κρυπτοκείμενο δεν φέρει καμία πληροφορία που να συνδέεται άμεσα ή έμμεσα με το αρχικό κείμενο, διασφαλίζοντας έτσι την απόλυτη μυστικότητα. Τέτοιες λύσεις κρυπτογράφησης που παρέχουν τέλεια μυστικότητα χρησιμοποιούνται συνήθως στις πιο ευαίσθητες επικοινωνίες.

Μεταξύ των αλγορίθμων κρυπτογράφησης ροής που προτείνονται για τρέχοντα συστήματα και μελλοντική χρήση, είναι οι HC-128, Salsa20/20 και η παραλλαγή του ChaCha, ο SNOW 2.0 και ο SNOW 3G, που χρησιμοποιούνται στις τηλεπικοινωνίες, καθώς και ο SOSEMANUK. Οι περισσότεροι από αυτούς τους αλγορίθμους κρυπτογράφησης υποβλήθηκαν για αξιολόγηση στον διαγωνισμό e-Stream, ένα έργο που εκτελέστηκε στην ΕΕ και είχε στόχο να εντοπίσει νέους αλγορίθμους κρυπτογράφησης ροής κατάλληλους για ευρεία υιοθέτηση. Ο Πίνακας 1.4 παρουσιάζει το σύνολο των αλγορίθμων κρυπτογράφησης ροής καθώς και την προτεινόμενη χρήση τους για συστήματα παλαιού τύπου (legacy systems) καθώς και για μελλοντικά συστήματα (future systems).

Πίνακας 1.4: Αλγόριθμοι κρυπτογράφησης ροής.

Αλγόριθμος	Κατηγοριοποίηση	
	Παλαιού Τύπου	Μελλοντικά Χρήση
HC-128	✓	✓
Salsa20/20	✓	✓
ChaCha	✓	✓
SNOW 2.0	✓	✓
SNOW 3G	✓	✓
SOSEMANUK	✓	✓
Grain	✓	✗
Grain-128a	✓	✗
Mickey 2.0	✓	✗
Trivium	✓	✗
Rabbit	✓	✗
AS/1	✗	✗
AS/2	✗	✗
E0	✗	✗
RC4	✗	✗

1.5.1 HC-128

Ο αλγόριθμος κρυπτογράφησης ροής HC-128 είναι είναι ένας απλός, ασφαλής, και αποδοτικός σε υλοποιήσεις λογισμικού, αλγόριθμος κρυπτογράφησης. Χρησιμοποιεί ένα κλειδί των 128 bit μαζί με ένα διάνυσμα αρχικοποίησης επίσης των 128 bit. Αποτελεί την απλοποιημένη έκδοση του HC-256 [12], ο οποίος χρησι-

μοποιεί κλειδιά των 256 bit και διανύσματα αρχικοποίησης ίδιου μεγέθους, ωστόσο δεν υποβλήθηκε στην αξιολόγηση eSTREAM.

Ο HC-128 χρησιμοποιεί δύο μυστικούς πίνακες, ο καθένας με 512 στοιχεία των 32 bit. Σε κάθε βήμα ενημερώνεται ένα στοιχείο ενός πίνακα με μια συνάρτηση μη γραμμικής ανάδρασης (non-linear feedback function). Ως αποτέλεσμα, όλα τα στοιχεία των δύο πινάκων ενημερώνονται κάθε 1024 βήματα. Σε κάθε βήμα, παράγεται μία έξοδος των 32 bit από τη συνάρτηση φίλτραρίσματος της μη γραμμικής εξόδου.

Ο HC-128 συμμετείχε στον διαγωνισμό eSTREAM και συμπεριλήφθηκε στο τελικό χαρτοφυλάκιο του eSTREAM ως πολλά υποσχόμενος για εφαρμογές λογισμικού [13].

Τρόπος Λειτουργίας: Ο HC-128 χρησιμοποιεί τις ακόλουθες λειτουργίες:

- + : το $x + y$ συμβολίζει το $x + y \bmod 2^{32}$, όπου $0 \leq x < 2^{32}$ and $0 \leq y < 2^{32}$
- \oplus : αποκλειστικό Ή
- $\|$: συνένωση
- \gg : τελεστής δεξιάς ολίσθησης (right shift operator) – $x \gg n$ σημαίνει ολίσθηση του x προς τα δεξιά κατά n bits
- \ll : τελεστής αριστερής ολίσθησης (left shift operator) – $x \ll n$ σημαίνει ολίσθηση του x προς τα αριστερά κατά n bits
- \ggg : τελεστής δεξιάς περιστροφής (right rotation operator) – $x \ggg n$ σημαίνει $((x \gg n) \oplus (x \ll (32 - n)))$, όπου $0 \leq n < 32$, $0 \leq x < 2^{32}$
- \lll : τελεστής αριστερής περιστροφής (left rotation operator) – $x \lll n$ σημαίνει $((x \ll n) \oplus (x \gg (32 - n)))$, όπου $0 \leq n < 32$, $0 \leq x < 2^{32}$

Το διάνυσμα αρχικοποίησης συμβολίζεται ως IV ενώ το κλειδί και η κλειδοροή που δημιουργείται, ως K και s αντίστοιχα. Ο HC-128 χρησιμοποιεί δύο πίνακες: P και Q όπως ορίζονται ακολούθως:

- P : ένας πίνακας αποτελούμενος από 512 στοιχεία των 32 bits και κάθε στοιχείο συμβολίζεται ως $P[i]$, όπου $0 \leq i < 512$
- Q : ένας πίνακας αποτελούμενος από 512 στοιχεία των 32 bits και κάθε στοιχείο συμβολίζεται ως $Q[i]$, όπου $0 \leq i < 512$
- K : το κλειδί μεγέθους 128 bits
- IV : το διάνυσμα αρχικοποίησης, μεγέθους 128 bits
- s : η κλειδοροή που δημιουργεί ο HC-128. Η 32-bit έξοδος που δημιουργεί ο HC-128 κατά το i -οστό βήμα, συμβολίζεται ως s_i . Επομένως $s = s_0 \| s_1 \| s_2 \| \dots$

Διαδικασία Αρχικοποίησης Κλειδιού και Διανύσματος Αρχικοποίησης: Η διαδικασία αρχικοποίησης του HC-128 αποτελείται από την επέκταση του κλειδιού και του διάνυσματος αρχικοποίησης στους πίνακες P και Q και εκτέλεση των 1024 βημάτων του αλγορίθμου κρυπτογράφησης (με τις εξόδους να χρησιμοποιούνται για την ενημέρωση των P και Q).

1. Έστω $K = K_0 \| K_1 \| K_2 \| K_3$ και $IV = IV_0 \| IV_1 \| IV_2 \| IV_3$, όπου κάθε K_i και IV_i υπονοεί έναν 32-bit αριθμό. Έστω $K_{i+4} = K_i$, και $IV_{i+4} = IV_i$ για $0 \leq i < 4$. Το κλειδί και το διάνυσμα αρχικοποίησης IV επεκτείνονται σε ένα διάνυσμα W_i ($0 \leq i \leq 1279$) ως ακολούθως:

$$W_i = \begin{cases} K_i & 0 \leq i \leq 7 \\ IV_{i-8} & 8 \leq i \leq 15 \\ f_2(W_{i-2}) + W_{i-7} + f_1(W_{i-15}) + W_{i-16} + i & 16 \leq i \leq 1279 \end{cases}$$

2. Ενημέρωση των πινάκων P και Q με το διάνυσμα W .

$$\begin{aligned} P[i] &= W_{i+256} && \text{για } 0 \leq i \leq 511 \\ Q[i] &= W_{i+768} && \text{για } 0 \leq i \leq 511 \end{aligned}$$

3. Εκτέλεση των βημάτων κρυπτογράφησης 1024 φορές και χρήση των εξόδων για την αντικατάσταση των στοιχείων του πίνακα ως εξής:

```

for  $i = 0 \dots 511$  do
     $P[i] = (P[i] + g_1(P[i - 3 \bmod 512], P[i - 10 \bmod 512], P[i - 511 \bmod 512]))$ 
     $\oplus h_1(P[i - 12 \bmod 512]);$ 
end

for  $i = 0 \dots 511$  do
     $Q[i] = (Q[i] + g_2(Q[i - 3 \bmod 512], Q[i - 10 \bmod 512], Q[i - 511 \bmod 512]))$ 
     $\oplus h_2(Q[i - 12 \bmod 512]);$ 
end

```

Με αυτόν τον τρόπο, η διαδικασία αρχικοποίησης ολοκληρώνεται και ο αλγόριθμος κρυπτογράφησης είναι έτοιμος να δημιουργήσει την απαιτούμενη για την κρυπτογράφηση κλειδοροή.

Αλγόριθμος Δημιουργίας Κλειδοροής: Σε κάθε βήμα, ένα στοιχείο ενός πίνακα ενημερώνεται και δημιουργείται μια έξοδος των 32 bits. Ο αλγόριθμος δημιουργίας της κλειδοροής στον HC-128 παρουσιάζεται στον Αλγόριθμο 1.5.

Αλγόριθμος 1.5: Δημιουργία κλειδοροής στον HC-128.

Είσοδος: $i = 0$;
Έξοδος: Bits κλειδοροής;

- 1 **while** απαιτούνται περισσότερα bits κλειδοροής **do**
- 2 $j = i \bmod 512$;
- 3 **if** $(i \bmod 1024) < 512$ **then**
- 4 $P[j] = P[j] + g_1(P[j - 3 \bmod 512], P[j - 10 \bmod 512], P[j - 511 \bmod 512]);$
- 5 $s_i = h_1(P[j - 12 \bmod 512]) \oplus P[j];$
- 6 **else**
- 7 $Q[j] = Q[j] + g_2(Q[j - 3 \bmod 512], Q[j - 10 \bmod 512], Q[j - 511 \bmod 512]);$
- 8 $s_i = h_2(Q[j - 12 \bmod 512]) \oplus Q[j];$
- 9 **end**
- 10 $i = i + 1$;
- 11 **end**

1.5.2 Salsa20/20 και ChaCha

Ο Salsa20/r ήταν ένας από τους συμμετέχοντες αλγορίθμους στον διαγωνισμό eSTREAM. Υποστηρίζει μήκη κλειδιών 128 και 256 bit. Η παράμετρος r αναφέρεται στον αριθμό των γύρων που χρησιμοποιούνται. Ο Salsa20/12 συμπεριλήφθηκε στο τελικό χαρτοφυλάκιο του eSTREAM ως ένας πολλά υποσχόμενος αλγόριθμος για εφαρμογές λογισμικού. Ο σχεδιαστής του Salsa20 συνιστά να χρησιμοποιείται το σύνολο των 20 γύρων.

Ο αλγόριθμος κρυπτογράφησης ροής ChaCha είναι μια παραλλαγή της οικογένειας Salsa20. Τροποποιεί τη σχεδίαση του Salsa για καλύτερη απόδοση και αυξημένη διάχυση. Ο αλγόριθμος κρυπτογράφησης ροής

ChaCha αποτελεί τη βάση του φιναλίστ BLAKE στον διαγωνισμό συνάρτησης σύνοψης SHA-3. Επίσης χρησιμοποιείται στο πρόγραμμα περιήγησης ιστού Google Chrome.

Οι Aumasson *et al.* [14] αναφέρουν μια επίθεση στον Salsa20/8 που απαιτεί 2^{251} κρυπτογραφήσεις και 2^{31} επιλεγμένα διανύσματα αρχικοποίησης. Αυτό ισχύει και για την οικογένεια ChaCha αλλά με υψηλότερο κόστος.

1.5.3 SNOW 2.0

Ο αλγόριθμος SNOW 2.0 περιλαμβάνεται στο πρότυπο ISO/IEC 18033-4 [15] και υποστηρίζει μεγέθη κλειδιών των 128 και 256 bits. Παρόλο που ο SNOW 2.0 θεωρείται ασφαλής για τις περισσότερες πρακτικές εφαρμογές, έχει αποδειχθεί θεωρητικά ότι μια διακριτική επίθεση εναντίον του είναι δυνατή. Ωστόσο, για την επιτυχή εκτέλεση αυτής της επίθεσης απαιτούνται 2^{174} bits από τη ροή κλειδιού και τεράστιοι υπολογιστικοί πόροι, γεγονός που την καθιστά μη πρακτική στην πράξη.

Επιπλέον, έχει ανακαλυφθεί μια επίθεση σχετικού κλειδιού όταν χρησιμοποιείται κλειδί των 256 bits, η οποία υπογραμμίζει την ανάγκη για προσοχή κατά την υλοποίηση του αλγορίθμου, ιδιαίτερα σε περιβάλλοντα όπου υπάρχει πιθανότητα εκμετάλλευσης τέτοιων αδυναμιών. Οι επιθέσεις σχετικού κλειδιού (related-key attacks) είναι μια μορφή κρυπτογραφικής επίθεσης κατά την οποία ο επιτιθέμενος έχει τη δυνατότητα να παρατηρεί ή να επιλέγει ζεύγη κλειδιών που σχετίζονται μεταξύ τους με κάποιον γνωστό τρόπο (π.χ. μέσω μιας απλής διαφοράς ή μετασχηματισμού). Αυτή η γνώση σχετικά με τη σχέση μεταξύ των κλειδιών επιτρέπει στον επιτιθέμενο να εκμεταλλευτεί τις αδυναμίες του κρυπτογραφικού αλγορίθμου, ώστε να ανακαλύψει πληροφορίες για τα κλειδιά.

1.5.4 SNOW 3G

Ο SNOW 3G αποτελεί μια βελτιωμένη εκδοχή του SNOW 2.0, με την πιο σημαντική αλλαγή να είναι η προσθήκη ενός δεύτερου S-box, που λειτουργεί ως προστασία ενάντια σε μελλοντικές εξελίξεις στην αλγεβρική κρυπτανάλυση. Η αλγεβρική κρυπτανάλυση είναι μια κρυπτογραφική τεχνική ανάλυσης που βασίζεται στη μαθηματική αναπαράσταση ενός κρυπτογραφικού αλγορίθμου ως ένα σύστημα αλγεβρικών εξισώσεων. Ο στόχος της αλγεβρικής κρυπτανάλυσης είναι να λύσει αυτές τις εξισώσεις για να αποκαλύψει κρυφές πληροφορίες, όπως το κλειδί κρυπτογράφησης ή το αρχικό κείμενο, εκμεταλλευόμενη τις αλγεβρικές ιδιότητες του αλγορίθμου.

Ο αλγόριθμος χρησιμοποιεί ένα κλειδί κρυπτογράφησης 128 bit και ένα διάνυσμα αρχικοποίησης IV, επίσης 128 bit. Ο SNOW 3G αποτελεί τον πυρήνα των αλγορίθμων UEA2 και UIA2, οι οποίοι χρησιμοποιούνται για την κρυπτογράφηση και την ακεραιότητα δεδομένων στο σύστημα 3GPP UMTS. Οι αλγόριθμοι αυτοί είναι πανομοιότυποι με τους 128-EEA1 και 128-EIA1, που εφαρμόζονται στις κρυπτογραφικές λειτουργίες του LTE.

1.5.5 SOSEMANUK

Ο SOSEMANUK [16] συμμετείχε στον διαγωνισμό eSTREAM¹ και συμπεριλήφθηκε στο τελικό χαρτοφυλάκιο του eSTREAM, καθώς θεωρήθηκε πολλά υποσχόμενος για εφαρμογές λογισμικού. Ο SOSEMANUK υποστηρίζει μήκη κλειδιών από 128 έως 256 bits, μαζί με ένα διάνυσμα αρχικοποίησης (IV) μήκους 128 bits. Οι σχεδιαστές του αλγορίθμου ισχυρίζονται ότι ο SOSEMANUK παρέχει επίπεδα ασφάλειας ισοδύναμα με αυτά των 128 bits (βλέπε Ορισμό 1.1), ανεξάρτητα από το μήκος του κλειδιού.

Στη βιβλιογραφία έχουν παρουσιαστεί διάφορες επιθέσεις κατά του SOSEMANUK, καμία από τις οποίες

¹Ο διαγωνισμός eSTREAM ήταν μια σημαντική πρωτοβουλία που διοργανώθηκε από το ECRYPT (European Network of Excellence in Cryptology) μεταξύ 2004 και 2008, με στόχο την εύρεση και προώθηση νέων αλγορίθμων κρυπτογράφησης ροής (stream ciphers) που είναι αποδοτικοί για διάφορες εφαρμογές, τόσο σε λογισμικό όσο και σε υλικό (hardware). Ο διαγωνισμός eSTREAM αποτέλεσε μέρος του ευρύτερου ερευνητικού προγράμματος ECRYPT, το οποίο είχε σκοπό να ενισχύσει την έρευνα και την ανάπτυξη κρυπτογραφικών πρωτοκόλλων και τεχνικών στην Ευρώπη.

όμως δεν έχει παραβιάσει τον ισχυρισμό της ασφάλειας των 128-bit. Μια επίθεση που απαιτούσε μόνο λίγες λέξεις από την κλειδοροή και θα χρειαζόταν έναν εξαιρετικά μεγάλο αριθμό υπολογισμών (περίπου 2^{176}) παρουσιάστηκε στο [17]. Επίσης, μια άλλη επίθεση που απαιτούσε περίπου 2^{138} λέξεις βασικής ροής και έναν τεράστιο αριθμό υπολογισμών (περίπου 2^{138}) παρουσιάστηκε στα [18, 19].

Ορισμός 1.1 (Επίπεδο Ασφάλειας). Το επίπεδο ασφάλειας, γνωστό και ως «bits ασφάλειας», είναι ένας αριθμός που σχετίζεται με την προσπάθεια που απαιτείται για την παραβίαση ενός κρυπτογραφικού αλγόριθμου ή συστήματος. Σύμφωνα με το NIST SP 800-57 [20], η ισχύς ασφαλείας καθορίζεται σε bit και είναι μια συγκεκριμένη τιμή από το σύνολο {80, 112, 128, 192, 256}. Αυτό επιτρέπει την εύκολη σύγκριση μεταξύ αλγορίθμων και είναι χρήσιμο όταν συνδυάζονται πολλαπλοί μηχανισμοί σε ένα υβριδικό κρυπτοσύστημα. Για παράδειγμα, ο αλγόριθμος AES-128 (μέγεθος κλειδιού 128 bit) έχει σχεδιαστεί για να προσφέρει ένα επίπεδο ασφάλειας 128 bit, το οποίο θεωρείται περίπου ισοδύναμο με ένα κρυπτοσύστημα RSA που χρησιμοποιεί κλειδί 3072 bit. Σημειώστε ότι η ισχύς ασφαλείας των 80 bit δεν θεωρείται πλέον επαρκώς ασφαλής.

1.5.6 Αλγόριθμοι Ροής Παλαιού Τύπου

1.5.6.1 Grain, Grain-v1 και Grain-128a

Ο Grain αποτελεί οικογένεια αλγορίθμων κρυπτογράφησης ροής. Η αρχική έκδοση είχε υποβληθεί στον διαγωνισμό eSTREAM [21]. Μετά την κρυπτανάλυση [22], η πρώτη έκδοση του Grain αναθεωρήθηκε σε Grain-v1 [23]. Η έκδοση Grain-v1 που υποστηρίζει ένα κλειδί 80 bit και ένα διάνυσμα αρχικοποίησης 64 bit συμπεριλήφθηκε στο τελικό χαρτοφυλάκιο eSTREAM ως πολλά υποσχόμενη για υλοποιήσεις υλικού. Ο Grain-128, ο οποίος είναι η έκδοση του Grain-v1 με κλειδί 128 bit και διάνυσμα προετοιμασίας 80 bit, δεν εγκρίνεται.

Μια ενημερωμένη έκδοση του Grain, που ονομάζεται Grain-128a, προτάθηκε στο [24] και οι προηγούμενες εκδόσεις για αυτήν την αναφορά το συνιστούσαν για μελλοντική χρήση. Μια πρόσφατη επίθεση από τους Todo *et al.* [25], μπορεί τώρα να καλύψει την πλήρη κρυπτογράφηση χρησιμοποιώντας δεδομένα 2113:8 σε χρόνο 2115:4. Ενώ αυτές οι πολυπλοκότητες παραμένουν απρόσιτες, η πλήρης επίθεση, καθώς και η παράλληλη πρόοδος στην κρυπτανάλυση άλλων μελών της οικογένειας Grain υποδηλώνουν αυξημένο κίνδυνο κατά τη χρήση του Grain-128a και για αυτό το λόγο δε προτείνεται για μελλοντική χρήση.

1.5.6.2 Mickey 2.0

Ο Mickey 2.0 αξιολογήθηκε από τον διαγωνισμό eSTREAM και συμπεριλήφθηκε στο τελικό χαρτοφυλάκιο του eSTREAM ως πολλά υποσχόμενος αλγόριθμος για υλοποιήσεις υλικού [26]. Χρησιμοποιεί ένα κλειδί 80-bit και ένα διάνυσμα αρχικοποίησης 80-bit. Υπάρχει επίσης μια κλιμακούμενη έκδοση Mickey-128 που χρησιμοποιεί κλειδιά και διάνυσμα αρχικοποίησης των 128 bits, αλλά αυτή η έκδοση δεν έχει αξιολογηθεί επίσημα από το eSTREAM [26].

1.5.6.3 Rabbit

Ο αλγόριθμος Rabbit συμμετείχε στον διαγωνισμό eSTREAM και συμπεριλήφθηκε στο τελικό χαρτοφυλάκιο eSTREAM ως πολλά υποσχόμενο για εφαρμογές λογισμικού. Ο Rabbit χρησιμοποιεί ένα κλειδί 128-bit μαζί με ένα IV 64-bit. Περιγράφεται στο RFC 4503 και περιλαμβάνεται στο ISO/IEC 18033-4 [15].

1.5.6.4 Trivium

Ο αλγόριθμος Trivium συμμετείχε στον διαγωνισμό eSTREAM και συμπεριλήφθηκε στο εθνικό χαρτοφυλάκιο eSTREAM ως πολλά υποσχόμενη για υλοποιήσεις υλικού. Έχει συμπεριληφθεί στο ISO/IEC 29192-3 [27] για την κατηγορία των αλγορίθμων ροής ελαφράς κρυπτογραφίας (Κεφάλαιο 13). Ο Trivium χρησιμοποιεί ένα κλειδί 80-bit μαζί με ένα IV 80-bit.

1.5.6.5 Μη ασφαλείς αλγόριθμοι κρυπτογράφησης ροής

A5/1: Ο A5/1 σχεδιάστηκε αρχικά για την κρυπτογράφηση επικοινωνιών στο πρωτόκολλο GSM και χρησιμοποιεί ένα κλειδί μήκους 64 bit, μαζί με έναν αριθμό πλαισίου 22 bit. Ο αριθμός πλαισίου (frame number) είναι μια ακολουθιακή τιμή που χρησιμοποιείται στα πρωτόκολλα επικοινωνίας, όπως το GSM, για να διασφαλιστεί η συγχρονισμένη και σωστή μετάδοση των δεδομένων μεταξύ του κινητού τηλεφώνου και του δικτύου.

Συγκεκριμένα, στο GSM, κάθε μετάδοση δεδομένων χωρίζεται σε μικρότερα τμήματα που ονομάζονται πλαίσια (frames), και ο αριθμός πλαισίου είναι ένας μετρητής που τα αριθμεί με μια ακολουθία 22 bit. Αυτός ο αριθμός είναι δημοσίως γνωστός και μεταδίδεται μαζί με τα δεδομένα για να βοηθήσει τόσο το δίκτυο όσο και τη συσκευή να παραμείνουν συγχρονισμένα.

Στον αλγόριθμο A5/1, ο αριθμός πλαισίου χρησιμοποιείται ως μέρος της αρχικοποίησης του αλγορίθμου κρυπτογράφησης μαζί με το κλειδί 64 bit. Αυτή η διαδικασία αρχικοποίησης είναι σημαντική για την κρυπτογράφηση της ροής δεδομένων. Δεδομένου ότι ο αριθμός πλαισίου είναι γνωστός και δεν αλλάζει συχνά, μπορεί να δώσει στους επιτιθέμενους ένα πλεονέκτημα όταν προσπαθούν να σπάσουν την κρυπτογράφηση, αφού μειώνει την πολυπλοκότητα του συστήματος.

Παρόλο που ο σχεδιασμός του A5/1 κρατήθηκε μυστικός κατά τα πρώτα χρόνια της ανάπτυξής του, το 1994 διέρρευσε το γενικό του σχέδιο. Έκτοτε, έχει αποκαλυφθεί πλήρως μέσω διεργασιών αντίστροφης μηχανικής, καθιστώντας όλες τις λεπτομέρειες του αλγορίθμου γνωστές στο ευρύ κοινό.

Ο A5/1 έχει γίνει στόχος πολλών κρυπτογραφικών επιθέσεων. Οι πιο αποτελεσματικές επιθέσεις έχουν αποδείξει ότι είναι δυνατόν να αποκρυπτογραφηθούν συνομιλίες κινητής τηλεφωνίας GSM σε πραγματικό χρόνο, εκμεταλλευόμενες αδυναμίες του αλγορίθμου [28]. Αυτές οι επιθέσεις έχουν καταδείξει τα σοβαρά κενά ασφαλείας του A5/1, με αποτέλεσμα να θεωρείται μη ασφαλής για την προστασία των τηλεπικοινωνιών σε σύγχρονα συστήματα GSM.

A5/2: Ο A5/2 αποτελεί μια σκόπιμα εξασθενημένη εκδοχή του αλγορίθμου A5/1, που σχεδιάστηκε για να ικανοποιεί τους περιορισμούς εξαγωγής κρυπτογραφικών τεχνολογιών σε ορισμένες χώρες, όπως ήταν τη δεκαετία του 1990. Η εξασθένηση αυτή επιβλήθηκε ώστε να επιτρέπεται η χρήση του αλγορίθμου σε διεθνείς αγορές, διασφαλίζοντας όμως παράλληλα ότι οι κρυπτογραφικές δυνατότητές του θα είναι ευάλωτες σε κυβερνήσεις ή άλλους φορείς με επαρκείς πόρους. Ως αποτέλεσμα, ο A5/2 έχει σοβαρά κενά ασφαλείας και μπορεί να παραβιαστεί με σχετικά απλές μεθόδους, καθιστώντας τον ακατάλληλο για χρήση σε σύγχρονες εφαρμογές όπου απαιτείται υψηλό επίπεδο ασφάλειας.

E0: Ο αλγόριθμος ροής E0 χρησιμοποιείται για την κρυπτογράφηση δεδομένων στα συστήματα Bluetooth και βασίζεται σε κλειδιά κρυπτογράφησης μήκους 128 bit. Σε αντίθεση με πολλούς άλλους αλγόριθμους ροής, ο E0 δεν χρησιμοποιεί αρχικό διανύσμα (IV) για να ενισχύσει την ασφάλεια, γεγονός που περιορίζει την πολυπλοκότητα των παραγόμενων κρυπτογραφημένων ροών. Η καλύτερη γνωστή επίθεση κατά τον E0 μπορεί να ανακτήσει το πλήρες κλειδί κρυπτογράφησης χρησιμοποιώντας τα πρώτα 24 bit από περίπου 2^{24} πλαίσια δεδομένων, σε συνδυασμό με περίπου 2^{38} υπολογισμούς [29]. Αυτή η επίθεση υποδηλώνει ότι η ασφάλεια του E0 είναι ανεπαρκής για την προστασία ευαίσθητων δεδομένων, καθώς απαιτεί σχετικά περιορισμένους πόρους για την επιτυχή αποκρυπτογράφηση. Ως αποτέλεσμα, ο E0 θεωρείται πλέον μη ασφαλής αλγόριθμος για σύγχρονες εφαρμογές.

RC4: Ο RC4 υποστηρίζει διάφορα μεγέθη κλειδιών. Παρά την ευρεία χρήση του αλγορίθμου είναι γνωστό εδώ και πολλά χρόνια ότι πάσχει από μια σειρά από αδυναμίες. Υπάρχουν διάφορες διακριτικές επιθέσεις [30] και επιθέσεις ανάκτησης κατάστασης [31]. Επιπρόσθετα, μια αποτελεσματική τεχνική για την ανάκτηση του μυστικού κλειδιού από μια εσωτερική κατάσταση περιγράφεται στο [32].

Ένα σημαντικό μειονέκτημα του RC4 είναι ότι αρχικά σχεδιάστηκε χωρίς την υποστήριξη διανύσματος αρχικοποίησης (IV). Ορισμένες εφαρμογές, όπως το WEP και το WPA, προσπάθησαν να αντισταθμίσουν αυτήν την έλλειψη χρησιμοποιώντας μέρος του κλειδιού ως IV. Ωστόσο, αυτή η πρακτική εισάγει νέες αδυναμίες, καθιστώντας δυνατές επιθέσεις σχετικού κλειδιού. Αυτή η προσέγγιση έχει οδηγήσει σε αποτελεσματικές επι-

θέσεις ανάκτησης κλειδιού, ιδιαίτερα στο WEP. Επιπλέον, όταν ο αλγόριθμος RC4 αρχικοποιείται, τα πρώτα 512 byte εξόδου θα πρέπει να απορριφθούν, λόγω ευπαθειών σχετικών με στατιστική ανάλυση. Εάν αυτό το βήμα αγνοηθεί, οι επιθέσεις ανάκτησης κλειδιού, όπως αυτές που στοχεύουν το WEP και το WPA, γίνονται πιο γρήγορες και αποδοτικές [33].

Παρά τα στατιστικά προβλήματα που είναι γνωστά από το 1995, το SSL/TLS δεν απορρίπτει κανένα από τα bytes εξόδου του RC4. Αυτό έχει ως αποτέλεσμα πρόσφατες επιθέσεις από τους AlFardan *et al.* [34] και Isobe *et al.* [35]. Βελτιωμένες επιθέσεις στο RC4 σε WPA-TKIP και TLS έχουν αναπτυχθεί από τους Vanhoef και Piessens [36].

1.6 Ασκήσεις-Εργασίες

Εργασίες

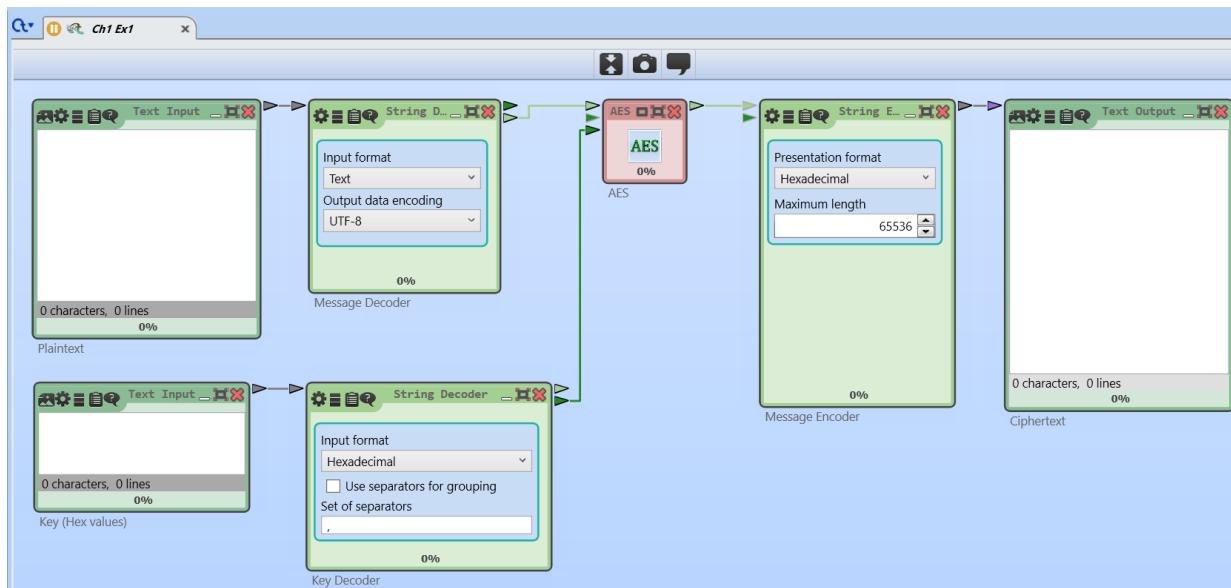
1.6.1 Χρησιμοποιώντας την εφαρμογή [CrypTool 2](#), υλοποιήστε το σενάριο που εμφανίζεται στο Σχήμα 1.15 για να κρυπτογραφήσετε κείμενο της επιλογής σας με τη χρήση του αλγορίθμου AES σε CBC και κλειδί κρυπτογράφησης μεγέθους 128 bit. Για την εξοικείωση σας με την εφαρμογή CrypTool 2 μπορείτε να δείτε το Παράρτημα A.

- (1) Καθώς το κλειδί που χρησιμοποιείτε δεν είναι απαραίτητα ένα ισχυρό κλειδί, θα μπορούσατε να αυξήσετε την ασφάλεια του κρυπτοσυστήματος σας δημιουργώντας ένα ισχυρό κλειδί με τη βοήθεια του στοιχείου PKCS#5.
 - Το PKCS#5 (*Public-Key Cryptography Standards #5*) [37] είναι ένα πρότυπο για την κρυπτογράφηση δεδομένων με τη χρήση αλγορίθμων συμμετρικής κρυπτογράφησης. Αναπτύχθηκε από την RSA Security και είναι μέρος της σειράς προτύπων PKCS για την κρυπτογραφία δημοσίου κλειδιού. Το PKCS#5 περιλαμβάνει τη μέθοδο PBKDF2 (*Password-Based Key Derivation Function 2*), η οποία χρησιμοποιείται για την παράγωγη κλειδιών από κωδικούς πρόσβασης. Αυτή η λειτουργία επιτρέπει τη δημιουργία ενός ισχυρού κλειδιού από έναν κωδικό πρόσβασης. Η διαδικασία περιλαμβάνει την επαναληπτική εφαρμογή μιας συνάρτησης σύνοψης σε έναν κωδικό πρόσβασης και μια τυχαία τιμή που ονομάζεται “salt”, βελτιώνοντας την ασφάλεια απέναντι σε επιθέσεις εξαντλητικής αναζήτησης (*brute-force*) και λεξικού (*dictionary attacks*).
- (2) Επεκτείνετε το κρυπτοσύστημα σας έτσι ώστε να κάνει και την απαιτούμενη για τον παραλήπτη του μηνύματος αποκρυπτογράφηση του κρυπτοειδένου (Προσοχή: Για την ορθή απεικόνιση του αποκρυπτογραφημένου κειμένου, θα χρειαστείτε επιπλέον ένα στοιχείο «String Encoder»).

1.6.2 Χρησιμοποιώντας την εφαρμογή [CrypTool 2](#) και το υπάρχον υλοποιημένο template AES Visualization, μελετήστε τον τρόπο λειτουργίας του αλγορίθμου AES.

Βιβλιογραφία

- [1] Hans Delfs and Helmut Knebl. “Symmetric-Key Cryptography”. In: *Introduction to Cryptography: Principles and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 11–48. ISBN: 978-3-662-47974-2. DOI: [10.1007/978-3-662-47974-2\2](https://doi.org/10.1007/978-3-662-47974-2_2).
- [2] Joan Daemen and Vincent Rijmen. *The design of Rijndael: AES — the Advanced Encryption Standard*. Springer-Verlag, 2002, p. 238. ISBN: 3-540-42580-2.



Σχήμα 1.15: Κρυπτογράφηση κειμένου με την χρήση του αλγορίθμου AES στο CrypTool 2.

- [3] NIST. *Specification for the Advanced Encryption Standard (AES)*. Federal Information Processing Standards Publication 197. 2001. URL: <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>.
- [4] ECRYPT. *D5.4 Algorithms, Key Size and Protocols Report*. Tech. rep. ECRYPT – Coordination & Support Action, 2018. URL: <https://www.ecrypt.eu.org/csa/publications.html>.
- [5] ENISA. *Algorithms, key size and parameters report – 2014*. Tech. rep. European Union Agency for Network and Information Security, Nov. 2014. URL: <https://www.enisa.europa.eu/publications/algorithms-key-size-and-parameters-report-2014>.
- [6] National Institute of Standards and Technology. *Advanced encryption standard (AES)*. Tech. rep. NIST FIPS 197. Gaithersburg, MD: National Institute of Standards and Technology, Nov. 2001, NIST FIPS 197. DOI: 10.6028/NIST.FIPS.197.
- [7] Joan Daemen and Vincent Rijmen. *The Design of Rijndael*. Ed. by Ueli Maurer et al. Information Security and Cryptography. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002. ISBN: 978-3-662-04722-4. DOI: 10.1007/978-3-662-04722-4.
- [8] Mitsuru Matsui, Shiho Moriai, and Junko Nakajima. *A Description of the Camellia Encryption Algorithm*. RFC 3713. Apr. 2004. DOI: 10.17487/RFC3713.
- [9] Eli Biham, Ross Anderson, and Lars Knudsen. “Serpent: A New Block Cipher Proposal”. In: *Fast Software Encryption*. Ed. by Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, and Serge Vaudenay. Vol. 1372. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 222–238. ISBN: 978-3-540-64265-7. DOI: 10.1007/3-540-69710-1_15.
- [10] Chanathip Namprempre, Tadayoshi Kohno, and Mihir Bellare. *The Secure Shell (SSH) Transport Layer Encryption Modes*. RFC 4344. Jan. 2006. DOI: 10.17487/RFC4344.
- [11] Phuong Ha Nguyen, Hongjun Wu, and Huaxiong Wang. “Improving the Algorithm 2 in Multi-dimensional Linear Cryptanalysis”. In: *Information Security and Privacy*. Ed. by Udaya Parampalli and Philip Hawkes. Vol. 6812. Series Title: Lecture Notes in Computer Science. Berlin, Heidel-

- berg: Springer Berlin Heidelberg, 2011, pp. 61–74. ISBN: 978-3-642-22497-3. DOI: 10.1007/978-3-642-22497-3_5.
- [12] Hongjun Wu. “A New Stream Cipher HC-256”. en. In: *Fast Software Encryption*. Ed. by Takeo Kanade et al. Vol. 3017. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 226–244. ISBN: 978-3-540-25937-4. DOI: 10.1007/978-3-540-25937-4_15.
- [13] Hongjun Wu. “The Stream Cipher HC-128”. en. In: *New Stream Cipher Designs*. Ed. by Matthew Robshaw and Olivier Billet. Vol. 4986. ISSN: 0302-9743, 1611-3349 Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 39–47. ISBN: 978-3-540-68351-3. DOI: 10.1007/978-3-540-68351-3_4.
- [14] Jean-Philippe Aumasson, Simon Fischer, Shahram Khazaei, Willi Meier, and Christian Rechberger. “New Features of Latin Dances: Analysis of Salsa, ChaCha, and Rumba”. en. In: *Fast Software Encryption*. Ed. by Kaisa Nyberg. Vol. 5086. ISSN: 0302-9743, 1611-3349 Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 470–488. ISBN: 978-3-540-71039-4. DOI: 10.1007/978-3-540-71039-4_30.
- [15] ISO/IEC 18033-4:2011. *Information technology — Security techniques — Encryption algorithms — Part 4: Stream ciphers*. International Organization for Standardization. 2011. URL: www.iso.org.
- [16] Côme Berbain et al. “Sosemanuk, a Fast Software-Oriented Stream Cipher”. en. In: *New Stream Cipher Designs*. Ed. by David Hutchison et al. Vol. 4986. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 98–118. ISBN: 978-3-540-68351-3. DOI: 10.1007/978-3-540-68351-3_9.
- [17] David Jao and Luca De Feo. “Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies”. In: *Post-Quantum Cryptography*. Ed. by Bo-Yin Yang. Vol. 7071. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 19–34. ISBN: 978-3-642-25405-5. DOI: 10.1007/978-3-642-25405-5_2.
- [18] Joo Yeon Cho and Miia Hermelin. “Improved Linear Cryptanalysis of SOSEMANUK”. In: *Proceedings of the 12th International Conference on Information Security and Cryptology*. ICISC’09. Seoul, Korea: Springer-Verlag, 2009, pp. 101–117. ISBN: 3642144225.
- [19] Jung-Keun Lee, Dong Hoon Lee, and Sangwoo Park. “Cryptanalysis of Sosemanuk and SNOW 2.0 Using Linear Masks”. In: *Advances in Cryptology - ASIACRYPT 2008*. Ed. by Josef Pieprzyk. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 524–538. ISBN: 978-3-540-89255-7.
- [20] Elaine Barker. *Recommendation for key management:: part 1 - general*. Tech. rep. NIST SP 800-57pt1r5. Gaithersburg, MD: National Institute of Standards and Technology, May 2020, NIST SP 800-57pt1r5. DOI: 10.6028/NIST.SP.800-57pt1r5.
- [21] Martin Hell, Thomas Johansson, and Willi Meier. “Grain: A Stream Cipher for Constrained Environments”. In: *Int. J. Wire. Mob. Comput.* 2.1 (May 2007), pp. 86–93. ISSN: 1741-1084. DOI: 10.1504/IJWMC.2007.013798.
- [22] Côme Berbain, Henri Gilbert, and Alexander Maximov. “Cryptanalysis of Grain”. In: *Fast Software Encryption*. Ed. by Matthew Robshaw. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 15–29. ISBN: 978-3-540-36598-3.
- [23] Martin Hell, Thomas Johansson, Alexander Maximov, and Willi Meier. “The Grain Family of Stream Ciphers”. In: *New Stream Cipher Designs: The eSTREAM Finalists*. Ed. by Matthew Robshaw and Olivier Billet. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 179–190. ISBN: 978-3-540-68351-3. DOI: 10.1007/978-3-540-68351-3_14.

- [24] Martin Ågren, Martin Hell, Thomas Johansson, and Willi Meier. “Grain-128a: A New Version of Grain-128 with Optional Authentication”. In: *Int. J. Wire. Mob. Comput.* 5.1 (Dec. 2011), pp. 48–59. ISSN: 1741-1084. doi: 10.1504/IJWMC.2011.044106.
- [25] Yosuke Todo, Takanori Isobe, Willi Meier, Kazumaro Aoki, and Bin Zhang. “Fast Correlation Attack Revisited”. In: *Advances in Cryptology – CRYPTO 2018*. Ed. by Hovav Shacham and Alexandra Boldyreva. Cham: Springer International Publishing, 2018, pp. 129–159. ISBN: 978-3-319-96881-0.
- [26] Steve Babbage and Matthew Dodd. “The MICKEY Stream Ciphers”. In: *New Stream Cipher Designs: The eSTREAM Finalists*. Ed. by Matthew Robshaw and Olivier Billet. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 191–209. ISBN: 978-3-540-68351-3. doi: 10.1007/978-3-540-68351-3_15.
- [27] ISO/IEC. *Information technology – Security techniques – Lightweight cryptography – Part 3: Stream ciphers*. Last accessed on August 26, 2024. International Organization for Standardization, 2012. URL: <https://www.iso.org/standard/56552.html>.
- [28] Elad Barkan, Eli Biham, and Nathan Keller. “Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication”. In: *Advances in Cryptology - CRYPTO 2003*. Ed. by Dan Boneh. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 600–616. ISBN: 978-3-540-45146-4.
- [29] Yi Lu, Willi Meier, and Serge Vaudenay. “The Conditional Correlation Attack: A Practical Attack on Bluetooth Encryption”. In: *Advances in Cryptology – CRYPTO 2005*. Ed. by Victor Shoup. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 97–117. ISBN: 978-3-540-31870-5.
- [30] Subhamoy Maitra and Goutam Paul. “New Form of Permutation Bias and Secret Key Leakage in Keystream Bytes of RC4”. In: *Fast Software Encryption*. Ed. by Kaisa Nyberg. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 253–269. ISBN: 978-3-540-71039-4.
- [31] Alexander Maximov and Dmitry Khovratovich. “New State Recovery Attack on RC4”. In: *Proceedings of the 28th Annual Conference on Cryptology: Advances in Cryptology*. CRYPTO 2008. Santa Barbara, CA, USA: Springer-Verlag, 2008, pp. 297–316. ISBN: 9783540851738. doi: 10.1007/978-3-540-85174-5_17.
- [32] Eli Biham and Yaniv Carmeli. “Efficient Reconstruction of RC4 Keys from Internal States”. In: *Fast Software Encryption*. Ed. by Kaisa Nyberg. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 270–288. ISBN: 978-3-540-71039-4.
- [33] Pouyan Sepehrdad, Serge Vaudenay, and Martin Vuagnoux. “Statistical Attack on RC4 Distinguishing WPA”. In: *Proceedings of the 30th Annual International Conference on Theory and Applications of Cryptographic Techniques: Advances in Cryptology*. EUROCRYPT’11. Tallinn, Estonia: Springer-Verlag, 2011, pp. 343–363. ISBN: 9783642204647.
- [34] Nadhem J. AlFardan, Daniel J. Bernstein, Kenneth G. Paterson, Bertram Poettering, and Jacob C. N. Schuldt. “On the Security of RC4 in TLS”. In: *Proceedings of the 22nd USENIX Conference on Security*. SEC’13. Washington, D.C.: USENIX Association, 2013, pp. 305–320. ISBN: 9781931971034.
- [35] Takanori Isobe, Toshihiro Ohigashi, Yuhei Watanabe, and Masakatu Morii. “Full Plaintext Recovery Attack on Broadcast RC4”. In: *Fast Software Encryption*. Ed. by Shiho Moriai. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 179–202. ISBN: 978-3-662-43933-3.
- [36] Mathy Vanhoef and Frank Piessens. “All Your Biases Belong to Us: Breaking RC4 in WPA-TKIP and TLS”. In: *Proceedings of the 24th USENIX Conference on Security Symposium*. SEC’15. Washington, D.C.: USENIX Association, 2015, pp. 97–112. ISBN: 9781931971232.
- [37] Burt Kaliski. *PKCS #5: Password-Based Cryptography Specification Version 2.0*. RFC 2898. Sept. 2000. doi: 10.17487/RFC2898.

ΚΕΦΑΛΑΙΟ 2

ΚΡΥΠΤΟΓΡΑΦΙΑ ΔΗΜΟΣΙΟΥ ΚΛΕΙΔΙΟΥ

Περίληψη

Η κρυπτογραφία δημοσίου κλειδιού [1], γνωστή και ως ασύμμετρη κρυπτογραφία, αποτελεί ένα άλλο σημαντικό βασικό στοιχείο στην κρυπτογραφία, που σε συνδυασμό με τους αλγορίθμους συμμετρικής κρυπτογραφίας, μπορεί να παρέχει μια σειρά πρόσθετων κρυπτογραφικών μηχανισμών στην ασφάλεια πληροφοριών. Οι αλγόριθμοι κρυπτογραφίας δημοσίου κλειδιού δε χρησιμοποιούνται για την κρυπτογράφηση μεγάλου όγκου δεδομένων. Ωστόσο αποτελούν μια ιδιαίτερα ελκυστική λύση για τη διαχείριση των κλειδιών συμμετρικής κρυπτογραφίας, καθώς και για τον πολύ χρήσιμο και με εκτεταμένη χρήση μηχανισμό των ψηφιακών υπογραφών. Με αυτόν τον τρόπο παρέχεται ακεραιότητα και αυθεντικοποίηση μηνυμάτων, καθώς επίσης και μη αποποίηση στις συναλλαγές, μια ιδιότητα που δεν μπορεί να παρασχεθεί από την συμμετρική κρυπτογραφία. Σε αυτό το κεφάλαιο αναλύονται οι βασικές αρχές και ιδιότητες της κρυπτογραφίας δημοσίου κλειδιού και παρουσιάζονται οι βασικότεροι αλγόριθμοι και οι τρόποι λειτουργίας αυτών. Πιο αναλυτικά, στην Ενότητα 2.1 γίνεται μια προσπάθεια ορισμού των βασικών αρχών που διέπουν τα κρυπτοσυστήματα δημοσίου κλειδιού, ενώ στις Ενότητες 2.2, 2.3 και 2.4 παρουσιάζονται τα κρυπτοσυστήματα δημοσίου κλειδιού RSA, ElGamal και Paillier, που παρουσιάζουν πληθώρα εφαρμογών στην σύγχρονη κρυπτογραφία. Στην Ενότητα 2.5 γίνεται περιγραφή του προτύπου ψηφιακών υπογραφών DSA που αποτελεί μια παραλλαγή των ψηφιακών υπογραφών του ElGamal. Στην Ενότητα 2.6 γίνεται μια εισαγωγή στην κρυπτογραφία ελειπτικών καμπυλών και παρουσιάζονται το σύστημα κρυπτογράφησης ECIES και το πρότυπο ψηφιακών υπογραφών ECDSA, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: –.

2.1 Ορισμός της Κρυπτογραφίας Δημοσίου Κλειδιού

Οι αλγόριθμοι της κρυπτογραφίας δημοσίου κλειδιού (Public-Key Cryptography – PKC), γνωστής και ως ασύμμετρης κρυπτογραφίας, σχεδιάστηκαν έτσι ώστε το κλειδί που χρησιμοποιείται για την κρυπτογράφηση να είναι διαφορετικό από το κλειδί που χρησιμοποιείται για την αποκρυπτογράφηση [2]. Επιπρόσθετα, το

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx-978-618-85370-x-x>

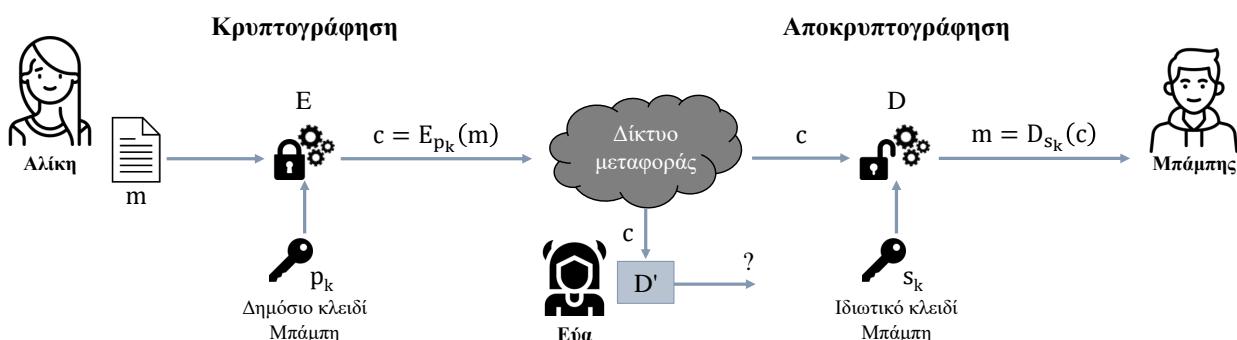
 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

κλειδί αποκρυπτογράφησης δεν μπορεί, τουλάχιστον σε ένα εύλογο χρονικό διάστημα, να υπολογιστεί από το κλειδί κρυπτογράφησης. Οι αλγόριθμοι αυτοί αναφέρονται ως «δημοσίου κλειδιού» επειδή το κλειδί κρυπτογράφησης μπορεί να γίνει δημόσια γνωστό προς όλους. Με αυτόν τον τρόπο, ένας άγνωστος μπορεί να χρησιμοποιήσει το κλειδί κρυπτογράφησης για να κρυπτογραφήσει ένα μήνυμα, αλλά μόνο ένα συγκεκριμένο πρόσωπο που είναι κάτοχος του αντίστοιχου κλειδιού αποκρυπτογράφησης μπορεί να αποκρυπτογραφήσει το μήνυμα (Σχήμα 2.1). Σε αυτά τα κρυπτοσυστήματα, το κλειδί κρυπτογράφησης καλείται συνήθως ως δημόσιο κλειδί, και το κλειδί αποκρυπτογράφησης καλείται ως ιδιωτικό κλειδί. Επίσης, το ιδιωτικό κλειδί μερικές φορές καλείται και ως μυστικό κλειδί, αλλά για να αποφευχθεί η σύγχυση με τους συμμετρικούς αλγόριθμους, αυτή η ορολογία γενικά αποφεύγεται [3]. Η πράξη της κρυπτογράφησης κάνει χρήση του δημόσιου κλειδιού p_k και εκφράζεται με την ακόλουθη εξίσωση:

$$E_{p_k}(m) = c$$

Αντίστοιχα, στην αποκρυπτογράφηση χρησιμοποιείται το ιδιωτικό κλειδί s_k και δίνεται από την ακόλουθη εξίσωση:

$$D_{s_k}(c) = m$$



Σχήμα 2.1: Διαδικασία κρυπτογράφησης και αποκρυπτογράφησης στα κρυπτοσυστήματα δημοσίου κλειδιού.

Ωστόσο, μερικές φορές, τα μηνύματα δύναται να κρυπτογραφούνται με το ιδιωτικό κλειδί και να αποκρυπτογραφούνται με το δημόσιο κλειδί, όπως συμβαίνει κατά την δημιουργία ψηφιακών υπογραφών στα κρυπτοσυστήματα δημοσίου κλειδιού [3]. Παρά την όποια πιθανή σύγχυση, αυτές οι διαδικασίες περιγράφονται με τις ακόλουθες δύο εξισώσεις:

$$E_{s_k}(m) = c$$

$$D_{p_k}(c) = m$$

Στα πλαίσια αυτού του βιβλίου, και για την αποφυγή των οποιωνδήποτε συγχύσεων, θα αποφευχθούν οι όροι κρυπτογράφηση και αποκρυπτογράφηση κατά την δημιουργία ψηφιακών υπογραφών. Αντιθέτως, θα γίνεται χρήση των όρων «δημιουργία υπογραφής» S_{s_k} και «επαλήθευση υπογραφής» V_{p_k} , αντίστοιχα.

2.2 Κρυπτοσύστημα RSA

Ένα από τα πιο δημοφιλή κρυπτογραφικά συστήματα δημοσίου κλειδιού είναι το κρυπτοσύστημα RSA [4], που προτάθηκε από τους Ronald Rivest, Adi Shamir και Leonard Adelman το 1977 και πήρε το όνομά του από τα αρχικά των επιθέτων τους. Μέχρι και σήμερα, το κρυπτοσύστημα RSA θεωρείται αδύνατο να σπάσει με τη χρήση σύγχρονων υπολογιστών, ωστόσο, εάν κάποια στιγμή οι κβαντικοί υπολογιστές αποκτήσουν

την ισχύ που τους αναλογεί, τότε ενδέχεται να αλλάξει ριζικά η ασφάλεια που παρέχεται, τόσο από τον RSA, όσο και από άλλα γνωστά κρυπτοσυστήματα δημοσίου κλειδιού [5]. Σήμερα, χρησιμοποιείται ευρύτατα στις υπηρεσίες που παρέχονται μέσω διαδικτύου, εξασφαλίζοντας εμπιστευτικότητα στις επικοινωνίες, καθώς και μη αποποίηση και αυθεντικοποίηση των πραγματοποιούμενων συναλλαγών.

2.2.1 Δημιουργία Κλειδιών

Η διαδικασία που πρέπει να ακολουθηθεί για τη δημιουργία ενός ζεύγους (δημόσιου και ιδιωτικού) κλειδιών RSA είναι η εξής:

1. Επιλέγουμε δύο τυχαίους μεγάλους πρώτους αριθμούς τον p και q και υπολογίζουμε το γινόμενο τους $n = p \cdot q$. Τα p και q κρατούνται μυστικά.
2. Με χρήση των p και q , υπολογίζουμε την συνάρτηση Euler $\Phi(n)$, που μας δίνει το πλήθος των θετικών ακεραίων που είναι μικρότεροι του n και (σχετικά) πρώτοι με το n :

$$\Phi(n) = \Phi(p) \cdot \Phi(q) = (p - 1)(q - 1)$$
3. Επιλέγουμε έναν τυχαίο αριθμό e μεγαλύτερο του 2 και μικρότερο του $\Phi(n)$ τέτοιο ώστε ο μέγιστος κοινός διαιρέτης $\gcd(e, \Phi(n)) = 1$. Για τον υπολογισμό του Μέγιστου Κοινού Διαιρέτη (ΜΚΔ) χρησιμοποιούμε τον αλγόριθμο του Ευκλείδη.
4. Το ζεύγος των αριθμών (e, n) αποτελεί το δημόσιο κλειδί p_k .
5. Με χρήση του γενικευμένου αλγόριθμου του Ευκλείδη, υπολογίζουμε τον ακέραιο αριθμό d , όπου $1 < d < \Phi(n)$, τέτοιον ώστε:

$$e \cdot d = 1 \pmod{\Phi(n)} \Leftrightarrow d \equiv e^{-1} \pmod{\Phi(n)}$$
6. Το ζεύγος των αριθμών (d, n) αποτελεί το ιδιωτικό κλειδί s_k .

Το δημόσιο κλειδί γίνεται διαθέσιμο προς όλους τους πιθανούς αποστολείς κρυπτογραφημένων μηνυμάτων ή επιβεβαιωτές ψηφιακών υπογραφών (π.χ. στον πελάτη της τράπεζας που θέλει να κάνει μία συναλλαγή μέσω του διαδικτύου). Το ιδιωτικό κλειδί το κρατά για την αποκλειστική του χρήση (για αποκρυπτογράφηση μηνυμάτων ή παραγωγή ψηφιακών υπογραφών) ο κάτοχος του (π.χ. η τράπεζα). Για παράδειγμα, ο αποστολέας κρυπτογραφεί το μήνυμα (π.χ. τα στοιχεία της συναλλαγής) με το δημόσιο κλειδί και στέλνει το μήνυμα στον παραλήπτη. Για να την αποκρυπτογράφηση του μηνύματος χρειάζεται το ιδιωτικό κλειδί. Το ιδιωτικό κλειδί δε απομακρύνεται ποτέ από την κατοχή του παραλήπτη, οπότε δεν υπάρχει κίνδυνος υποκλοπής.

2.2.2 Αλγόριθμος Κρυπτογράφησης

Ο αλγόριθμος που ακολουθείται για την κρυπτογράφηση RSA ενός μηνύματος m , $0 \leq m < n$, όπου m ακέραιος ($m \in Z_n$), είναι ο εξής:

1. Το δημόσιο κλειδί $p_k = (e, n)$ στέλνεται στον αποστολέα του μηνύματος.
2. Ο αποστολέας κρυπτογραφεί το μήνυμα m με βάση αυτό το δημόσιο κλειδί.
3. Το κρυπτογραφημένο μήνυμα $E_{p_k}(m) = m^e \pmod{n}$ αποστέλλεται στον παραλήπτη.

2.2.3 Αλγόριθμος Αποκρυπτογράφησης

Ο αλγόριθμος που ακολουθείται για την αποκρυπτογράφηση RSA ενός κρυπτοκειμένου c είναι ο εξής:

1. Το ιδιωτικό κλειδί $s_k = (d, n)$ παραμένει στην πλευρά του παραλήπτη.
2. Για να αποκρυπτογραφηθεί το κρυπτοκείμενο c χρειάζεται το ιδιωτικό κλειδί.
3. Το αποτέλεσμα της αποκρυπτογράφησης θα είναι $D_{s_k}(c) = c^d \pmod{n}$.

2.2.4 Δημιουργία και Επαλήθευση Υπογραφών

Οι ψηφιακές υπογραφές παρέχουν την δυνατότητα επαλήθευσης της προέλευσης ενός μηνύματος από ένα συγκεκριμένο αποστολέα, καθώς και της αυθεντικότητας του μηνύματος. Το κρυπτοσύστημα RSA μπορεί να χρησιμοποιηθεί για τη δημιουργία της υπογραφής ενός μηνύματος με χρήση του ιδιωτικού κλειδιού του αποστολέα και με επακόλουθη χρήση του δημοσίου κλειδιού του αποστολέα από τον παραλήπτη για την επαλήθευση της υπογραφής. Ωστόσο, αυτό προϋποθέτει ότι ο παραλήπτης έχει στην διάθεση του, εκτός από την υπογραφή s , και το ίδιο το μήνυμα m .

Ο αλγόριθμος που ακολουθείται για την δημιουργία υπογραφής RSA ενός μηνύματος m είναι ο εξής:

1. Ο αποστολέας χρησιμοποιεί το ιδιωτικό του κλειδί $s_k = (d, n)$ που βρίσκεται στην κατοχή του.
2. Ο αποστολέας υπογράφει το μήνυμα m κάνοντας χρήση του ιδιωτικού του κλειδιού.
3. Το αποτέλεσμα της υπογραφής θα είναι $s = S_{s_k}(m) = m^d \pmod{n}$.

Αντίστοιχα, ο αλγόριθμος που ακολουθείται για την επαλήθευση υπογραφής RSA ενός μηνύματος m είναι ο εξής:

1. Το δημόσιο κλειδί $p_k = (e, n)$ του αποστολέα είναι δημόσια διαθέσιμο και βρίσκεται στην πλευρά του παραλήπτη.
2. Ο παραλήπτης επαληθεύει την υπογραφή s για το μήνυμα m με βάση το δημόσιο κλειδί του αποστολέα.
3. Η υπογραφή είναι έγκυρη, εάν και μόνο εάν $V_{p_k}(m, s) = s^e \pmod{n} \equiv m$.

Ωστόσο, η δημιουργία υπογραφής που περιγράφεται παραπάνω θα πρέπει να γίνεται προσεκτικά για την αποφυγή σχετικών επιθέσεων [6]. Στην πράξη, προκειμένου να συμπιεστεί το μέγεθος μηνύματος, χρησιμοποιείται μια ασφαλής συνάρτηση σύνοψης $H : \{0,1\}^* \rightarrow Z_n^*$ (περισσότερα στο Κεφάλαιο 3) και αντί του μηνύματος m . Οπότε, οι παραπάνω δύο εξισώσεις δημιουργίας και επαλήθευσης μιας υπογραφής μετασχηματίζονται ως εξής:

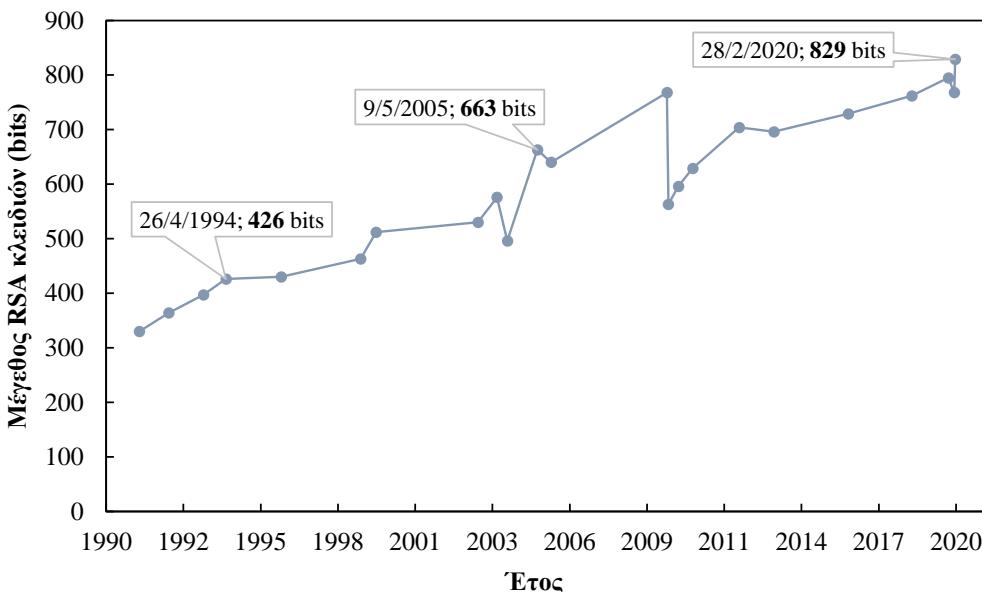
- Δημιουργία υπογραφής: $S_{s_k}(m) = (H(m))^d \pmod{n}$
- Επαλήθευση υπογραφής: $V_{p_k}(m, s) = s^e \pmod{n} \equiv H(m)$

2.2.5 Ασφάλεια του RSA

Για να καταλάβουμε καλύτερα την ασφάλεια [7] που μας παρέχει το κρυπτοσύστημα RSA αναφερθούμε πρώτα στο τι μπορούμε να κάνουμε για να αποκρυπτογραφήσουμε ένα μήνυμα που κρυπτογραφήθηκε με το κρυπτοσύστημα RSA, αλλά χωρίς να έχουμε στην κατοχή μας το ιδιωτικό κλειδί. Για να γίνει κάτι τέτοιο, θα πρέπει πρώτα να έχουμε στην διάθεση μας το δημόσιο κλειδί p_k που αντιστοιχεί σε αυτό το ιδιωτικό κλειδί, δηλαδή το ζεύγος των αριθμών (e, n) . Αφού τώρα γνωρίζουμε τον αριθμό n , δεν έχουμε παρά να τον αναλύσουμε σε γινόμενο δύο πρώτων αριθμών, έτσι ώστε να βρούμε τους αριθμούς p και q . Μόλις τους βρούμε, ο υπολογισμός του d και επομένως η αποκρυπτογράφηση μπορούν να γίνουν άμεσα, αφού η μέθοδος του κρυπτοσύστηματος RSA είναι γνωστή.

Ενώ είναι πολύ εύκολο να πολλαπλασιάσουμε δύο πρώτους αριθμούς για να βρούμε το γινόμενό τους, είναι πάρα πολύ δύσκολο να αναλύσουμε έναν αριθμό σε γινόμενο δύο πρώτων αριθμών και είναι πρακτικά αδύνατον αν ο αριθμός αυτός αποτελείται από πολλά ψηφία. Έτσι λοιπόν, οι πρώτοι αριθμοί p και q θα πρέπει να είναι αρκετά μεγάλοι, ώστε ο καλύτερος γνωστός αλγόριθμος παραγοντοποίησης να απαιτεί χρόνο μεγαλύτερο από αυτόν για τον οποίο θέλουμε να προστατευθούν τα κρυπτογραφημένα μηνύματα.

Οι Rivest, Shamir και Adelman, για να αποδείξουν ότι το κρυπτογραφικό τους σύστημα δεν μπορεί να σπάσει, ζήτησαν από όποιο νομίζει ότι μπορεί, να αναλύσει σε γινόμενο δύο πρώτων αριθμών μια λίστα από



Σχήμα 2.2: Μεγέθη RSA κλειδιών που έχουν παραγοντοποιηθεί από το 1991 μέχρι και το 2020 (Πηγή: [Wikipedia](#)).

διάφορα μεγέθη RSA τιμών του n μεταξύ 100 και 617 δεκαδικών ψηφίων (330 – 2048 bits, αντίστοιχα). Στο Σχήμα 2.2 παρουσιάζει διάφορα μεγέθη RSA κλειδιών που έχουν παραγοντοποιηθεί μέχρι και το 2020. Ενδεικτικά, το 1994 παραγοντοποιήθηκε ένας ακέραιος αριθμός με 129 ψηφία (426 bits) κάνοντας χρήση ενός δικτύου 1600 υπολογιστών με περίπου 600 εθελοντές συνδεδεμένους μέσω διαδικτύου [8], το 2005 παραγοντοποιήθηκε ένας αριθμός με 200 ψηφία (663 bits) από παράλληλους υπολογιστές που ισοδυναμούσαν με 75 χρόνια λειτουργίας ενός επεξεργαστή (2.2 GHz AMD Opteron) [9], ενώ το 2020 παραγοντοποιήθηκε ένας αριθμός με 250 ψηφία (829 bits) αξιοποιώντας 2700 χρόνια ενός επεξεργαστή (2.1 GHz Intel Xeon Gold 6130) [10]. Έτσι λοιπόν, ακόμη και με τα σημερινά τεχνολογικά δεδομένα, το πρόβλημα της ανάλυσης ενός αριθμού σε γινόμενο δύο πρώτων αριθμών είναι πρακτικά αδύνατον να λυθεί με τη χρήση σύγχρονων υπολογιστών και πόσο μάλλον όταν ο αριθμός αυτός αποτελείται από πολλά ψηφία.

Η αρχική έκδοση κρυπτογράφησης του RSA, που παρουσιάστηκε στις προηγούμενες υποενότητες, είναι ένας ντετερμινιστικός αλγόριθμος κρυπτογράφησης (δηλαδή, δεν έχει κάποιο τυχαίο στοιχείο) και ένας επιτιθέμενος μπορεί να πραγματοποιήσει μια κρυπταναλυτική επίθεση σε επιλεγμένο αρχικό κείμενο (CPA) (Ορισμός 2.1) κρυπτογραφώντας διάφορα αρχικά κείμενα της επιλογής του και εξετάζοντας αν είναι ίδια με το κρυπτοκείμενο που έχει στην κατοχή του. Ένα κρυπτοσύστημα χαρακτηρίζεται ως σημασιολογικά ασφαλές [11] εάν ένας επιτιθέμενος δεν μπορεί να διακρίνει δύο κρυπτοκείμενα μεταξύ τους, ακόμη και αν γνωρίζει (ή έχει επιλέξει) τα αντίστοιχα αρχικά κείμενα. Όπως γίνεται σαφές, το αρχικό κρυπτοσύστημα RSA δεν είναι σημασιολογικά ασφαλές. Η λύση σε αυτό το πρόβλημα ασφάλειας είναι ο padded-RSA (PKCS #1 v2.2) [12] που προσθέτει ψηφία τυχαιοποίησης r (τυχαία bits) σε ένα μήνυμα m , με τέτοιο τρόπο ώστε να είναι αναστρέψιμα μετά την αποκρυπτογράφηση. Έτσι, με αξιοποίηση αυτής της τεχνικής πλήρωσης (padding) φαίνεται ότι ο RSA μπορεί να εξασφαλίσει σημασιολογική ασφάλεια.

Ορισμός 2.1 (Κατηγορίες Κρυπταναλυτικών Επιθέσεων). Οι πιο γνωστές κατηγορίες κρυπταναλυτικών επιθέσεων είναι οι εξής:

- Επίθεση Μόνο σε Κρυπτοκείμενο (Ciphertext-Only Attack – COA): Αποτελεί ένα μοντέλο επίθεσης όπου ο (παθητικός) επιτιθέμενος θεωρείται ότι έχει πρόσβαση μόνο σε ένα σύνολο κρυπτοκειμένων.
- Επίθεση σε Γνωστό Αρχικό Κείμενο (Known-Plaintext Attack – KPA): Αποτελεί ένα μοντέλο επίθεσης όπου ο (παθητικός) επιτιθέμενος έχει πρόσβαση τόσο στο αρχικό κείμενο όσο και στην κρυπτο-

γραφημένη του έκδοση.

- Επίθεση σε Επιλεγμένο Αρχικό Κείμενο (Chosen-Plaintext Attack – CPA): Αποτελεί ένα μοντέλο επίθεσης που προϋποθέτει ότι ο (ενεργός) επιτιθέμενος μπορεί να λάβει κρυπτοκείμενα για αυθαίρετα αριθμό αρχικών κειμένων που ο ίδιος επιλέγει.
- Επίθεση σε Επιλεγμένο Κρυπτοκείμενο (Chosen-Ciphertext Attack – CCA): Αποτελεί ένα μοντέλο επίθεσης όπου ο (ενεργός) επιτιθέμενος μπορεί να συλλέξει πληροφορίες αποκτώντας πρόσβαση σε αποκρυπτογραφημένα κείμενα επιλεγμένων κρυπτοκειμένων.

2.3 Κρυπτοσύστημα ElGamal

Το κρυπτοσύστημα ElGamal παρέχει κρυπτογραφία δημοσίου κλειδιού που βασίζεται στην ιδέα των Diffie-Hellman [13] και προτάθηκε από τον Taher Elgamal το 1985 [14]. Η ανθεκτικότητα της κρυπτογράφησης του ElGamal βασίζεται στο πρόβλημα λύσης του διακριτού λογάριθμου. Στις υποενότητες που ακολουθούν γίνεται αναλυτική περιγραφή των αλγορίθμων δημιουργίας κλειδιών, κρυπτογράφησης, αποκρυπτογράφησης, δημιουργίας/επαλήθευσης υπογραφών, καθώς και η παρεχόμενη ασφάλεια που παρέχει.

2.3.1 Δημιουργία Κλειδιών

Η διαδικασία που πρέπει να ακολουθηθεί για τη δημιουργία ενός ζεύγους (δημόσιου και ιδιωτικού) κλειδιών ElGamal είναι η εξής:

1. Επιλέγουμε ένα τυχαίο μεγάλο και πρώτο αριθμό p και το Z_p^* υποδηλώνει το σύνολο όλων των ακέραιων $\{1, 2, \dots, p - 1\}$.
2. Επιλέγουμε ένα πρωταρχικό στοιχείο των γεννήτορα g από το σύνολο Z_p^* , τέτοιο ώστε $g^k \neq 1 \pmod{p}$ για όλα τα k μικρότερα του $p - 1$. Η διαδικασία εύρεσης του γεννήτορα g πραγματοποιείται ως εξής:
 - (α) Εάν $g^k = 1 \pmod{p}$ είναι αληθές για κάποιον ακέραιο αριθμό k στο διάστημα $1 \leq k \leq p - 1$, τότε ο αριθμός k διαιρέται του $p - 1$.
 - (β) Εάν θέλουμε να ελέγξουμε αν ένας αριθμός g είναι γεννήτορας \pmod{p} , δεν χρειάζεται να τον υψώσουμε σε όλες τις δυνάμεις $\{1, 2, \dots, p - 1\}$, αλλά είναι αποδοτικότερο να τον υψώσουμε μόνο στους διαιρέτες του $p - 1$.
3. Επιλέγουμε ένα τυχαίο αριθμό a στο διάστημα $1 \leq a \leq p - 1$ ως το ιδιωτικό κλειδί s_k .
4. Υπολογίζουμε το $y = g^a \pmod{p}$.
5. Το δημόσιο κλειδί p_k αποτελείται από τους αριθμούς (p, g, y) .

2.3.2 Αλγόριθμος Κρυπτογράφησης

Ο αλγόριθμος που ακολουθείται για την κρυπτογράφηση ElGamal ενός μηνύματος είναι ο εξής:

1. Επιλέγουμε έναν τυχαίο αριθμό r στο σύνολο $\{1, 2, \dots, p - 1\}$.
2. Εκφράζουμε το μήνυμα ως έναν ακέραιο αριθμό m στο σύνολο $\{1, 2, \dots, p - 1\}$.
3. Υπολογίζουμε το $\gamma = g^r \pmod{p}$ και το $\delta = m \cdot y^r \pmod{p}$ κάνοντας χρήση του δημοσίου κλειδιού $p_k = (p, g, y)$.
4. Το κρυπτογραφημένο μήνυμα αποτελείται από το ζεύγος $E_{p_k}(m, r) = (\gamma, \delta)$.

2.3.3 Αλγόριθμος Αποκρυπτογράφησης

Ο αλγόριθμος που ακολουθείται για την αποκρυπτογράφηση ElGamal ενός κρυπτοκειμένου (γ, δ) είναι ο εξής:

1. Υπολογίζουμε το γ^{-a} κάνοντας χρήση του ιδιωτικού κλειδιού $s_k = a$.
2. Το αρχικό μήνυμα θα είναι το $m = (\gamma^{-a}) \cdot \delta \pmod{p}$. Με άλλα λόγια η ανάκτηση του αρχικού μηνύματος γίνεται με την πράξη $\frac{\delta}{\gamma^a}$.
3. Το αποκρυπτογραφημένο κείμενο δηλαδή είναι το $D_{s_k}(\gamma, \delta) = m \pmod{p}$.

2.3.4 Δημιουργία και Επαλήθευση Υπογραφών

Ο αλγόριθμος που ακολουθείται από τον υπογράφοντα για την δημιουργία υπογραφής ElGamal ενός μηνύματος m είναι ο εξής:

1. Επιλέγει έναν τυχαίο αριθμό k , όπου $k \in Z_{p-1}^*$, τέτοιο ώστε ο μέγιστος κοινός διαιρέτης $\gcd(k, p - 1) = 1$. Το τυχαία επιλεγμένο k πρέπει να παραμείνει κρυφό και να είναι διαφορετικό για διαφορετικές υπογραφές.
2. Υπολογίζει το $r = g^k \pmod{p}$ και το $s = (m - a \cdot r) k^{-1} \pmod{p-1}$, όπου a το ιδιωτικό κλειδί s_k .
3. Η υπογραφή αποτελείται από το ζεύγος αριθμών $S_{s_k}(m, k) = (r, s)$.

Αντίστοιχα, ο αλγόριθμος που ακολουθείται από τον παραλήπτη για την επαλήθευση υπογραφής ElGamal ενός μηνύματος m είναι ο εξής:

1. Το δημόσιο κλειδί $p_k = (p, g, y)$ του υπογράφοντα είναι δημόσια διαθέσιμο και βρίσκεται στην πλευρά του παραλήπτη.
2. Ο παραλήπτης επαληθεύει ότι $0 < r < p$ και το $0 < s < p - 1$.
3. Η υπογραφή είναι έγκυρη, εάν και μόνο εάν $V_{p_k}(m, (r, s)) = y^r \cdot r^s \pmod{p} \equiv g^m$.

Ωστόσο, η υπογραφή που προκύπτει από την παραπάνω διαδικασία επιτρέπει μια μορφή επίθεσης, γνωστής ως υπαρξιακή πλαστογραφία (existential forgery), όπως αναφέρεται από τον ίδιο τον ElGamal (Ενότητα IV στο [14]). Για το λόγο αυτό, ενδείκνυται η χρήση μιας ασφαλούς συνάρτησης σύνοψης $H : \{0,1\}^* \rightarrow Z_p^*$ (περισσότερα στο Κεφάλαιο 3) πριν την υπογραφή του μηνύματος m (όπως ενδεικτικά αναφέρεται στο [15]). Οπότε, οι παραπάνω εξισώσεις δημιουργίας και επαλήθευσης μιας υπογραφής μετασχηματίζονται ως εξής:

- Δημιουργία υπογραφής: $S_{s_k}(m, k) = (g^k \pmod{p}, (H(m) - a \cdot r) k^{-1} \pmod{p-1})$
- Επαλήθευση υπογραφής: $V_{p_k}(m, (r, s)) = y^r \cdot r^s \pmod{p} \equiv g^{H(m)}$

2.3.5 Ασφάλεια του ElGamal

Ο επιτιθέμενος που θα επιχειρήσει μια επίθεση στο κρυπτοσύστημα ElGamal [16] και γνωρίζει το δημόσιο κλειδί $p_k = (p, g, y)$, θα πρέπει να ανακτήσει το ιδιωτικό κλειδί $s_k = a$, από τη εξίσωση:

$$y = g^a \mod p$$

Θα πρέπει δηλαδή να υπολογίσει τον διακριτό λογάριθμο με βάση g . Επομένως, θεωρούμε ότι η ασφάλεια του κρυπτοσυστήματος ElGamal βασίζεται στην επίλυση του προβλήματος του διακριτού λογαρίθμου, και η πιθανή εύρεση μιας τέτοιας λύσης θα καθιστούσε το κρυπτοσύστημα ανασφαλές.

Η ύπαρξη του τυχαίου αριθμού r κατά την κρυπτογράφηση, έχει ως αποτέλεσμα τη δυνατότητα αντιστοίχησης του αρχικού κειμένου σε $p - 1$ κρυπτοκείμενα. Η διαδικασία όπου το αρχικό κείμενο συνδυάζεται με μια τυχαία μεταβλητή, ονομάζεται διεργασία τυχαιοποίησης (randomization process). Το βήμα αυτό, το οποίο δεν υπάρχει στον RSA, καθιστά το κρυπτοσύστημα ElGamal ανθεκτικότερο σε επιθέσεις παρόμοιες με αυτές που παρουσιάζονται στον RSA, δηλαδή επιτυγχάνει σημασιολογική ασφάλεια σε επιθέσεις με επιλεγμένο αρχικό κείμενο (CPA), αλλά δεν είναι ασφαλές σε επιθέσεις με επιλεγμένο κρυπτοκείμενο (CCA) (Ορισμός 2.1). Βέβαια, η χρήση του τυχαίου αριθμού εισάγει έναν επιπλέον κίνδυνο που οδηγεί σε μια επιπρόσθετη απαίτηση. Για κάθε μήνυμα που κρυπτογραφείται, θα πρέπει να επιλέγεται διαφορετικός τυχαίος r . Στην περίπτωση που δύο μηνύματα m και m' κρυπτογραφηθούν με τον ίδιο r , τότε για τα αντίστοιχα κρυπτοκείμενα που θα προκύψουν (γ, δ) και (γ', δ') , η γνώση του ενός μηνύματος επιτρέπει την ανάκτηση του άλλου με βάση την ακόλουθη εξίσωση:

$$\frac{\delta}{\delta'} = \frac{m \cdot y^r}{m' \cdot y^r} = \frac{m}{m'}$$

Τέλος, κατά την κρυπτογράφηση με το κρυπτοσύστημα ElGamal, το μέγεθος των παραμέτρων αποτελεί σημαντικό κριτήριο υλοποίησης, λόγω του αυξημένου χρόνου που απαιτείται για την κρυπτογράφηση (δύο πράξεις ύψωσης σε δύναμη έναντι της μιας στην περίπτωση του RSA), καθώς και λόγω του διπλάσιου μεγέθους του κρυπτοκειμένου. Τα μειονεκτήματα αυτά έχουν σαν αποτέλεσμα να προτιμούνται κλειδιά μικρότερου μεγέθους σε σύγκριση με τον RSA, χωρίς ωστόσο να υπάρχουν εκπτώσεις ασφαλείας με αυτή την επιλογή.

2.4 Κρυπτοσύστημα Paillier

Το κρυπτοσύστημα Paillier προτάθηκε από τον Pascal Paillier το 1999 [17] και αποτελεί έναν πιθανοτικό ασύμμετρο αλγόριθμο κρυπτογράφησης δημόσιου κλειδιού. Το πρόβλημα στο οποίο βασίζεται, δηλαδή στον υπολογισμό κλάσεων νιοστών υπολοίπων (n -th residue classes), πιστεύεται ότι είναι υπολογιστικά δύσκολο. Η υπόθεση δυσκολίας στην οποία βασίζεται αυτό το κρυπτοσύστημα είναι η λεγόμενη Υπόθεση Αποφασιστικής Σύνθετης Υπολειματικότητας (Decisional Composite Residuosity Assumption – DCRA). Στις επόμενες υποενότητες παρέχεται μια λεπτομερής περιγραφή των αλγορίθμων δημιουργίας κλειδιών, κρυπτογράφησης και αποκρυπτογράφησης, καθώς και η παρεχόμενη ασφάλεια αυτού του αλγορίθμου.

2.4.1 Δημιουργία Κλειδιών

Η διαδικασία που πρέπει να ακολουθηθεί για τη δημιουργία ενός ζεύγους (δημόσιου και ιδιωτικού) κλειδιών του Paillier είναι η εξής:

1. Επιλέγουμε δύο μεγάλους τυχαίους πρώτους αριθμούς p και q , τέτοιους ώστε ο μέγιστος κοινός διαιρέτης $\gcd(p \cdot q, (p - 1)(q - 1)) = 1$.
2. Υπολογίζουμε το $n = p \cdot q$ και το $\lambda = \text{lcm}(p - 1, q - 1)$.
3. Επιλέγουμε έναν ακέραιο τυχαίο αριθμό g όπου $g \in Z_{n^2}^*$.

4. Εξασφαλίζουμε ότι το n διαιρεί τη τάξη του g ελέγχοντας την ύπαρξη του ακόλουθου πολλαπλασιαστικού αντίστροφου: $\mu = (L(g^\lambda \mod n^2))^{-1} \mod n$, όπου η συνάρτηση L ορίζεται ως $L(u) = \frac{u-1}{n}$.
5. Το δημόσιο κλειδί είναι το $p_k = (n, g)$ και το ιδιωτικό είναι $s_k = (\lambda, \mu)$.

2.4.2 Αλγόριθμος Κρυπτογράφησης

Ο αλγόριθμος που ακολουθείται για την κρυπτογράφηση Paillier ενός μηνύματος m με το δημόσιο κλειδί p_k είναι ο εξής:

1. Δεδομένου ότι m να είναι ένα μήνυμα προς κρυπτογράφηση, όπου $m \in Z_n$.
2. Επιλέγουμε έναν τυχαίο αριθμό r , όπου $r \in Z_n^*$.
3. Το κρυπτογραφημένο μήνυμα υπολογίζεται ως εξής: $E_{p_k}(m, r) = g^m \cdot r^n \mod n^2$.

2.4.3 Αλγόριθμος Αποκρυπτογράφησης

Ο αλγόριθμος που ακολουθείται για την αποκρυπτογράφηση Paillier ενός κρυπτοκειμένου c είναι ο εξής:

1. Έστω c το κρυπτοκείμενο του κρυπτογραφημένου μηνύματος m , όπου $c \in Z_{n^2}^*$.
2. Για την αποκρυπτογράφηση του μηνύματος, χρειάζεται μόνο το ιδιωτικό κλειδί $s_k = (\lambda, \mu)$.
3. Το αποτέλεσμα της αποκρυπτογράφησης είναι $D_{s_k}(c) = L(c^\lambda \mod n^2) \cdot \mu \mod n$.

2.4.4 Ασφάλεια του Paillier

Το κρυπτοσύστημα Paillier [17], όπως παρουσιάστηκε παραπάνω, παρέχει σημασιολογική ασφάλεια [11] σε επιθέσεις επιλεγμένου αρχικού κειμένου (CPA). Η ικανότητα να διακρίνει κανείς επιτυχώς το κρυπτοκείμενο ουσιαστικά ισοδυναμεί με την ικανότητα να αποφασίζει τη σύνθετη υπολειματικότητα (composite residuosity). Η ασφάλειά του βασίζεται στην λεγόμενη Υπόθεση Αποφασιστικής Σύνθετης Υπολειματικότητας (Decisional Composite Residuosity Assumption – DCRA) η οποία πιστεύεται ότι είναι δυσεπίλυτη.

Ωστόσο, λόγω της ομοιορφικής ιδιότητας που παρουσιάζει (περισσότερα στην Ενότητα 8.2), το κρυπτοσύστημα είναι εύπλαστο (malleable) και επομένως δεν διασφαλίζει το υψηλότερο κλιμάκιο σημασιολογικής ασφάλειας που προστατεύει από επιθέσεις επιλεγμένου κρυπτοκειμένου (CCA) (Ορισμός 2.1). Συνήθως, στην κρυπτογραφία ή έννοια της ευπλαστότητας (malleability) δεν θεωρείται πλεονέκτημα, αλλά σε ορισμένες εφαρμογές, όπως η ασφαλής ηλεκτρονική ψηφοφορία [18] και τα κρυπτοσύστημα κατωφλίου (threshold) [19], η ιδιότητα αυτή μπορεί πράγματι να είναι απαραίτητη.

Ωστόσο, οι Paillier και Pointcheval [20] πρότειναν στη συνέχεια ένα βελτιωμένο κρυπτοσύστημα το οποίο ενσωματώνει τον υπολογισμό της σύνοψης (hashing) του μηνύματος m με έναν τυχαίο αριθμό r . Αυτή η σύνοψη εμποδίζει έναν επιτιθέμενο, ο οποίος έχει γνώση μόνο του κρυπτοκειμένου c , να είναι σε θέση να αλλάξει το αποκρυπτογραφημένο μήνυμα m με έναν τρόπο που να έχει νόημα. Μέσω αυτής της προσαρμογής, το βελτιωμένο κρυπτοσύστημα Paillier μπορεί να αποδειχθεί ότι είναι ασφαλές ακόμη και σε επιθέσεις επιλεγμένου κρυπτοκειμένου (CCA).

2.5 Αλγόριθμος Ψηφιακών Υπογραφών DSA

Ο αλγόριθμος ψηφιακών υπογραφών μήπως Ψηφιακής Υπογραφής;) DSA (Digital Signature Algorithm) αποτελεί ένα πρότυπο ψηφιακών υπογραφών του FIPS (Federal Information Processing Standards), βασι- σμένο στη μαθηματική έννοια της εκθετοποίησης υπολοίπων (modular exponentiation) και στο πρόβλημα του διακριτού λογάριθμου. Ο αλγόριθμος DSA ουσιαστικά αποτελεί μια παραλλαγή του συστήματος υπο- γραφών του ElGamal (Ενότητα 2.3.4), ωστόσο με αρκετά μικρότερο μέγεθος υπογραφής.

Το NIST (National Institute of Standards and Technology) πρότεινε τον αλγόριθμο DSA για χρήση στο πρότυπο DSS (Digital Signature Standard) το 1991 και το νιοθέτησε ως FIPS 186 το 1994 [21]. Ο DSA είναι κατοχυρωμένος με δίπλωμα ευρεσιτεχνίας, αλλά το NIST έχει καταστήσει αυτήν την πατέντα διαθέσιμη παγκοσμίως χωρίς δικαιώματα. Στις υποενότητες που ακολουθούν γίνεται αναλυτική περιγραφή των αλγο- ρίθμων δημιουργίας κλειδιών και δημιουργίας/επαλήθευσης υπογραφών.

2.5.1 Δημιουργία Κλειδιών

Η διαδικασία που πρέπει να ακολουθηθεί για τη δημιουργία ενός ζεύγους (δημόσιου και ιδιωτικού) κλειδιών DSA είναι η εξής:

1. Επιλέγουμε μια ασφαλή συνάρτηση σύνοψης H με έξοδο $|H|$ bits. Στην αρχική έκδοση του DSA [21], η συνάρτηση σύνοψης H ήταν η SHA-1, αλλά στην 4^η έκδοση του DSA [22] προτείνεται η SHA-2 (περισσότερο στο Κεφάλαιο 3). Εάν η έξοδος της συνάρτησης σύνοψης $|H|$ είναι μεγαλύτερη σε μήκος από ένα μέγεθος μόντουλο N , χρησιμοποιούνται μόνο τα αριστερά N bits της συνάρτησης σύνοψης.
2. Επιλέγουμε μέγεθος κλειδιού L bits. Η αρχική έκδοση του DSA [21] περιόριζε το L να είναι πολλα- πλάσιο του 64 μεταξύ 512 και 1024 bits. Ωστόσο, το NIST το 2007 [23] συνιστά μέγεθος κλειδιών 2048 και 3072 bits, ώστε η διάρκεια ανθεκτικότητας να υπερβαίνει το 2010 και το 2030, αντίστοιχα [23].
3. Επιλέγουμε το μέγεθος μόντουλο N τέτοιο ώστε $N < L$ και $N \leq |H|$. Η 4^η έκδοση του DSA [22] καθορίζει το L και το N να έχουν μία από τις ακόλουθες τιμές (1024, 160), (2048, 224), (2048, 256) ή (3072, 256).
4. Επιλέγουμε έναν πρώτο αριθμό q μεγέθους N -bits.
5. Επιλέγουμε έναν πρώτο αριθμό p μεγέθους L -bits τέτοιο ώστε το $p - 1$ να είναι πολλαπλάσιο του q .
6. Επιλέγουμε τυχαία έναν ακέραιο αριθμό h από το σύνολο $\{2, \dots, p - 2\}$.
7. Υπολογίζουμε τον γεννήτορα g ως εξής $g = h^{(p-1)/q} \pmod p$. Στην περίπτωση που το $g = 1$ δοκιμά- ζουμε ξανά με διαφορετικό h . Συνήθως χρησιμοποιείται το $h = 2$. Αυτή η μόντουλο ύψωση σε δύναμη μπορεί να υπολογιστεί αποτελεσματικά ακόμη και αν οι τιμές είναι μεγάλες.
8. Επιλέγουμε ως ιδιωτικό κλειδί s_k έναν τυχαίο ακέραιο x από το σύνολο $\{1, \dots, q - 1\}$.
9. Υπολογίζουμε το $y = g^x \pmod p$ και το δημόσιο κλειδί p_k θα αποτελείται από τους αριθμούς (p, q, g, y) .

Στην παραπάνω διαδικασία, τα βήματα 1-7 μπορεί να είναι κοινά μεταξύ διαφορετικών χρηστών, ενώ τα βήματα 8-9 πρέπει να υπολογίζονται για κάθε χρήστη ξεχωριστά.

2.5.2 Δημιουργία και Επαλήθευση Υπογραφών

Ο αλγόριθμος που ακολουθείται από τον υπογράφοντα για την δημιουργία υπογραφής DSA ενός μηνύματος m είναι ο εξής:

1. Επιλέγει τυχαία έναν ακέραιο k από το σύνολο $\{1, \dots, q-1\}$. Το τυχαία επιλεγμένο k πρέπει να παραμείνει κρυφό και να είναι διαφορετικό για διαφορετικές υπογραφές.
2. Υπολογίζει το $r = (g^k \text{ mod } p) \text{ mod } q$. Στην σπάνια περίπτωση που το $r = 0$, επαναλαμβάνεται ο υπολογισμός με διαφορετικό τυχαίο k .
3. Υπολογίζει το $s = (k^{-1}(H(m) + x \cdot r)) \text{ mod } q$, όπου x το ιδιωτικό κλειδί s_k . Στην σπάνια περίπτωση που το $s = 0$, επαναλαμβάνεται ο υπολογισμός με διαφορετικό τυχαίο k .
4. Η υπογραφή αποτελείται από το ζεύγος αριθμών $S_{s_k}(m, k) = (r, s)$.

Αντίστοιχα, ο αλγόριθμος που ακολουθείται από τον παραλήπτη για την επαλήθευση υπογραφής DSA ενός μηνύματος m είναι ο εξής:

1. Το δημόσιο κλειδί $p_k = (p, q, g, y)$ του υπογράφοντα είναι δημόσια διαθέσιμο και βρίσκεται στην πλευρά του παραλήπτη.
2. Ο παραλήπτης επαληθεύει ότι $0 < r < q$ και $0 < s < q$.
3. Υπολογίζει το $e_1 = H(m) \cdot s^{-1} \text{ mod } q$.
4. Υπολογίζει το $e_2 = r \cdot s^{-1} \text{ mod } q$.
5. Η υπογραφή είναι έγκυρη, εάν και μόνο εάν $V_{p_k}(m, (r, s)) = (g^{e_1} \cdot y^{e_2} \text{ mod } p) \text{ mod } q \equiv r$.

2.6 Κρυπτογραφία Ελλειπτικών Καμπυλών

Η κρυπτογραφία ελλειπτικών καμπυλών (Elliptic-Curve Cryptography – ECC) [24] αποτελεί μια μορφή κρυπτογραφίας δημόσιου κλειδιού που αξιοποιεί την αλγεβρική δομή των ελλειπτικών καμπυλών σε πεπερασμένα σώματα (finite fields). Η ασφάλεια της κρυπτογραφίας ελλειπτικών καμπυλών βασίζεται στο πρόβλημα του διακριτού λογάριθμου των ελλειπτικών καμπυλών (Elliptic Curve Discrete Logarithm Problem – ECDLP), δηλαδή ότι η εύρεση του διακριτού λογάριθμου ενός τυχαίου στοιχείου ελλειπτικής καμπύλης σε σχέση με ένα δημόσια γνωστό σημείο-βάση είναι πρακτικά ανέφικτη. Η ασφάλεια που παρέχει αυτό το πρόβλημα εξαρτάται από την ικανότητα υπολογισμού ενός πολλαπλασιασμού σημείων και την αδυναμία υπολογισμού του πολλαπλασιαστή δεδομένου του αρχικού και του γινομένου των σημείων. Το μέγεθος της ελλειπτικής καμπύλης, μετρούμενο από το συνολικό αριθμό διακριτών ακεραίων ζευγών που ικανοποιούν την εξίσωση καμπύλης, καθορίζει τη δυσκολία του προβλήματος. Το πρόβλημα ECDLP είναι αρκετά πιο δύσκολο από τα μαθηματικά προβλήματα (δηλ. παραγοντοποίηση ακεραίων και διακριτός λογάριθμος) που βασίζονται τα υπόλοιπα κρυπτοσυστήματα δημόσιου κλειδιού, με αποτέλεσμα να απαιτείται αρκετά μικρότερο μέγεθος κλειδιών [25]. Μια συγκριτική παρουσίαση μεταξύ του μεγέθους κλειδιών RSA και ECC παρουσιάζεται στον Πίνακα 2.1 [23], όπου το επίπεδο ασφάλειας αντιπροσωπεύει την ασφάλεια που παρέχεται από έναν συμμετρικό αλγόριθμο κρυπτογράφησης χρησιμοποιώντας ένα κλειδί n -bits.

Η αξιοποίηση ελλειπτικών καμπυλών για κρυπτογραφική χρήση προτάθηκε για πρώτη φορά το 1985, σχεδόν ταυτόχρονα και ανεξάρτητα, από τους Neal Koblitz [26] και Victor Miller [27]. Έκτοτε, οι ελλειπτικές καμπύλες έχουν βρει εφαρμογές σε σχήματα συμφωνίας κλειδιού (key agreement), ψηφιακών υπογραφών, γεννητριών ψευδοτυχαίων αριθμών, κτλ. Εμμεσα, μπορούν επίσης να χρησιμοποιηθούν για κρυπτογράφηση συνδυάζοντας σχήματα συμφωνίας κλειδιού με συμμετρικά συστήματα κρυπτογράφησης.

Πίνακας 2.1: Συγκριτική παρουσίαση μεταξύ του μεγέθους κλειδιών RSA και ECC.

Επίπεδο Ασφάλειας (bits)	Μέγεθος RSA Κλειδιών (bits)	Μέγεθος ECC Κλειδιών (bits)
80	1024	160 – 223
112	2048	224 – 255
128	3072	256 – 283
192	7680	384 – 511
256	15360	512 – 571

2.6.1 Υπόβαθρο Ελλειπτικών Καμπυλών

2.6.1.1 Τύποι Ελλειπτικών Καμπυλών

Μια ελλειπτική καμπύλη E πάνω σε ένα πεπερασμένο σώμα (ή σώμα Galois) GF ορίζεται από την ακόλουθη εξίσωση, γνωστή ως εξίσωση Weierstrass για ελλειπτικές καμπύλες σε μη ομοιογενή μορφή [28]:

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_5 \quad (2.1)$$

όπου $a_1, a_2, a_3, a_4, a_5 \in GF$ και $\Delta \neq 0$, με Δ να αποτελεί την διακρίνουσα της E και η οποία υπολογίζεται με τον ακόλουθο τρόπο [29]:

$$\Delta = -d_1^2d_4 - 8d_2^3 - 27d_3^2 + 9d_1d_2d_3$$

όπου $d_1 = a_1^2 + 4a_2$, $d_2 = 2a_4 + a_1a_3$, $d_3 = a_3^2 + 4a_5$, και $d_4 = a_1^2a_5 + 4a_2a_5 - a_1a_3a_4 + a_2a_3^2 - a_4^2$. Η συνθήκη $\Delta \neq 0$ διασφαλίζει ότι η καμπύλη δεν είναι μοναδική και επομένως δεν υπάρχουν σημεία καμπύλης με δύο ή περισσότερες διαφορετικές εφαπτόμενες ευθείες.

Η ομοιογενής μορφή της εξίσωσης Weierstrass είναι:

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_5Z^3$$

και συνεπάγεται την ύπαρξη ενός ειδικού σημείου που μπορεί να ερμηνευθεί μόνο στο προβολικό επίπεδο, δηλαδή το σημείο στο άπειρο O . Το σημείο O είναι το σημαντικότερο στη χρήση ελλειπτικών καμπυλών στην κρυπτογραφία, καθώς είναι το στοιχείο ταυτότητας που, μαζί με τα υπόλοιπα σημεία της ελλειπτικής καμπύλης και τον τελεστή πρόσθεσης (που επιτρέπει την πρόσθεση δύο σημείων της ελλειπτικής καμπύλης, P και Q , προκειμένου να δημιουργηθεί ένα άλλο σημείο $R = P + Q$), χαρακτηρίζει την ελλειπτική καμπύλη με τη μαθηματική δομή της αβελιανής ομάδας.

Όταν το ίδιο σημείο προστίθεται πολλές φορές στον εαυτό του στην αβελιανή ομάδα που ορίζεται από μια ελλειπτική καμπύλη, ο τελεστής πρόσθεσης μετατρέπεται σε κλιμακωτό πολλαπλασιασμό, ο οποίος στην πράξη επιτρέπει τον πολλαπλασιασμό ενός σημείου P της ελλειπτικής καμπύλης με έναν θετικό ακέραιο αριθμό n με σκοπό να παράγει ένα άλλο σημείο $S = n \cdot P$ της ελλειπτικής καμπύλης.

Ο αριθμός των σημείων μιας ελλειπτικής καμπύλης (έννοια γνωστή και ως καρδινάλιος ή η τάξη της καμπύλης) αναπαρίσταται ως $\#E$. Αντίθετα, η τάξη ενός σημείου P που ανήκει σε μια ελλειπτική καμπύλη E είναι ο μικρότερος ακέραιος n που έχει ως αποτέλεσμα $n \cdot P = O$.

Από κρυπτογραφικής πλευράς, κάθε ελλειπτική καμπύλη δεν είναι χρήσιμη. Στην κρυπτογραφία ενδιαφέρομαστε για ελλειπτικές καμπύλες που σχηματίζουν κυκλικές αβελιανές ομάδες, καθώς και για ελλειπτικές καμπύλες με κυκλικές υποομάδες, έτσι ώστε ο συμπαράγοντας (cofactor) να είναι ένας μικρός αριθμός (π.χ. 2, 4, κτλ.). Ως αποτέλεσμα του θεωρήματος Lagrange (το οποίο δηλώνει ότι για οποιαδήποτε πεπερασμένη ομάδα M , η τάξη κάθε υποομάδας N της M διαιρεί την τάξη του M), η τάξη του γεννήτορα (δηλ. το σημείο της ελλειπτικής καμπύλης που παράγει όλα τα σημεία της κυκλικής υποομάδας) διαιρεί πάντα τη τάξη της ελλειπτικής καμπύλης (η οποία δεν είναι απαραίτητα πρώτος αριθμός).

Δύο τύποι πεπερασμένων σωμάτων $GF(q)$, με $q = p^m$ στοιχεία, χρησιμοποιούνται στην κρυπτογραφία ελλειπτικών καμπυλών: τα πρώτα πεπερασμένα σώματα $GF(p)$ (όπου p είναι περιττός πρώτος αριθμός και $m = 1$) και τα δυαδικά πεπερασμένα σώματα $GF(2^m)$ (όπου $p = 2$ και m μπορεί να είναι οποιοσδήποτε ακέραιος αριθμός μεγαλύτερος του 1). Όταν γίνεται χρήση πεπερασμένων σωμάτων, με την κατάλληλη αλλαγή μεταβλητών, είναι δυνατόν να απλοποιηθεί η εξίσωση Weierstrass, δημιουργώντας νέες εξισώσεις λιγότερο γενικές (προσαρμοσμένες σε συγκεκριμένα πεπερασμένα σώματα) αλλά ευκολότερες στη διαχείριση.

Εάν θεωρήσουμε ότι το πεπερασμένο σώμα έχει χαρακτηριστική $p = 2$ (δηλ. είναι δυαδικό), τότε $GF(q) = GF(2^m)$, και εάν το $a_1 \neq 0$, η εξίσωση 2.1 μπορεί να απλοποιηθεί στη μορφή:

$$y^2 + xy = x^3 + ax^2 + b, \quad \Delta = b \quad (2.2)$$

Τώρα, εάν το $a_1 = 0$, η εξίσωση 2.1 μπορεί να μετατραπεί στη μορφή:

$$y^2 + cy = x^3 + ax + b, \quad \Delta = c^4 \quad (2.3)$$

Επιπλέον, εάν το πεπερασμένο σώμα έχει χαρακτηριστική $p = 3$, τότε εμφανίζονται δύο περιπτώσεις. Εάν το $a_1^2 \neq -a_2$, η εξίσωση 2.1 γίνεται:

$$y^2 = x^3 + ax^2 + b, \quad \Delta = -a^3b \quad (2.4)$$

Αντίθετα, εάν το $a_1^2 = -a_2$, τότε η εξίσωση 2.1 γίνεται:

$$y^2 = x^3 + ax + b, \quad \Delta = -a^3 \quad (2.5)$$

Τέλος, εάν το πεπερασμένο σώμα έχει χαρακτηριστική p διάφορη του 2 ή του 3, χρησιμοποιώντας την κατάλληλη αλλαγή των μεταβλητών στην εξίσωση 2.1 αυτή μπορεί να μετατραπεί ως εξής:

$$y^2 = x^3 + ax + b, \quad \Delta = -16(4a^3 + 27b^2) \quad (2.6)$$

2.6.1.2 Παράμετροι Τομέα

Το σύνολο των παραμέτρων τομέα (domain parameters), που θα χρησιμοποιηθούν σε κάθε εφαρμογή κρυπτογραφίας ελλειπτικών καμπυλών, εξαρτάται από το υποκείμενο πεπερασμένο σώμα. Όταν το σώμα είναι $GF(p)$, το σύνολο των παραμέτρων που καθορίζουν την καμπύλη είναι (p, a, b, G, n, h) , ενώ αν το πεπερασμένο σώμα είναι $GF(2^m)$, το σύνολο των παραμέτρων είναι $(m, f(x), a, b, G, n, h)$. Η έννοια της κάθε παραμέτρου και στα δύο πεπερασμένα σώματα είναι η ακόλουθη:

- Το p είναι ο πρώτος αριθμός που χαρακτηρίζει το πεπερασμένο σώμα $GF(p)$.
- Το m είναι ο ακέραιος αριθμός που καθορίζει το πεπερασμένο σώμα $GF(2^m)$.
- Η συνάρτηση $f(x)$ είναι το πολυώνυμο βαθμού m που ορίζει το σώμα $GF(2^m)$.
- Τα a και b είναι τα στοιχεία του πεπερασμένου σώματος $GF(q)$ που υπάρχουν στις εξισώσεις 2.2 – 2.6.
- Το $G = (G_x, G_y)$ είναι το σημείο της καμπύλης που θα χρησιμοποιηθεί ως γεννήτορας των σημείων της καμπύλης που αντιπροσωπεύουν τα δημόσια κλειδιά.
- Το n είναι ο πρώτος αριθμός του οποίου η τιμή αντιπροσωπεύει τη τάξη του σημείου G (δηλ. $n \cdot G = O$).
- Το h είναι ο συμπαράγοντας (cofactor) της καμπύλης, που υπολογίζεται ως $h = \#E/n$, όπου n είναι η τάξη του γεννήτορα G .

Η δημιουργία παραμέτρων τομέα δεν γίνεται συνήθως από τον κάθε ενδιαφερόμενο ξεχωριστά, διότι αυτό περιλαμβάνει τον υπολογισμό του αριθμού των σημείων σε μια καμπύλη, η οποία είναι χρονοβόρα διαδικασία και δεν είναι εύκολα υλοποιήσιμη. Ως αποτέλεσμα, αρκετοί φορείς προτυποποίησης έχουν δημοσιεύσει παραμέτρους τομέα ελλειπτικών καμπυλών για αρκετά κοινά μεγέθη σωμάτων. Τέτοιες παράμετροι τομέα είναι κοινώς γνωστές ως «πρότυπες καμπύλες» και είναι διαθέσιμες από φορείς όπως το NIST [22], το SEC [30], και το ECC Brainpool [31].

2.6.2 Κρυπτογραφικό Σχήμα ECIES

Το πιο διαδεδομένο κρυπτογραφικό σχήμα που βασίζεται στην κρυπτογραφία ελλειπτικών καμπυλών είναι το ECIES (Elliptic Curve Integrated Encryption Scheme). Αυτό το σχήμα αποτελεί μια παραλλαγή του κρυπτοσυστήματος ElGamal και προτάθηκε το 1999 από τους Abdalla, Bellare και Rogaway [32]. Ελαφρώς διαφορετικές εκδόσεις του ECIES παρατίθενται στα πρότυπα ANSI X9.63 [33], IEEE 1363a [34], ISO/IEC 18033-2 [35] και SEC G 1 [24]. Το ECIES αποτελεί ουσιαστικά ένα υβριδικό σχήμα κρυπτογραφίας που παρέχει σημασιολογική ασφάλεια [11] έναντι ενός επιτιθέμενου ο οποίος επιτρέπεται να χρησιμοποιεί επιθέσεις επιλεγμένου αρχικού κειμένου (CPA) και επιλεγμένου κρυπτοκειμένου (CCA) (Ορισμός 2.1). Η ασφάλεια του κρυπτογραφικού αυτού σχήματος βασίζεται στο υπολογιστικό πρόβλημα Diffie-Hellman (Diffie-Hellman Problem – DHP) [13], με βάση το οποίο αν κάποιος γνωρίζει το $g^x \text{ mod } p$ και το $g^y \text{ mod } p$ είναι υπολογιστικά αδύνατο να βρει το $g^{xy} \text{ mod } p$ (όπου p πρώτος αριθμός και g γεννήτορας στο \mathbb{Z}_p^*).

2.6.2.1 Προαπαιτούμενα Κρυπτογράφησης

Για να σταλεί ένα κρυπτογραφημένο μήνυμα χρησιμοποιώντας το κρυπτογραφικό σχήμα ECIES, προαπαιτούνται τα ακόλουθα:

- Να οριστούν οι κρυπτογραφικές τεχνικές που θα χρησιμοποιηθούν για:
 - Παραγωγή κλειδιών (Key Derivation Function – KDF): Αποτελεί μια κρυπτογραφική συνάρτηση σύνοψης που δημιουργεί ένα σύνολο κλειδιών βάση μιας μυστικής πληροφορίας (όπως λειτουργεί ο αλγόριθμος ανταλλαγής κλειδιών Diffie-Hellman). Παράδειγμα τέτοιας συνάρτησης είναι η ANSI-X9.63-KDF [33].
 - Παραγωγή κώδικα αυθεντικοποίησης μηνύματος (Message Authentication Code – MAC): Αποτελεί μια σύντομη πληροφορία που χρησιμοποιείται για τον έλεγχο της ακεραιότητας ενός μηνύματος. Παραδείγματα τέτοιων τεχνικών είναι η HMAC-SHA-1-160 με μέγεθος κλειδιών 160 bits και η HMAC-SHA-1-80 με μέγεθος κλειδιών 80 bits (Ενότητα 5.2).
 - Εφαρμογή συμμετρικής κρυπτογραφίας: Συγκεκριμένα, καθορίζεται ο αλγόριθμος συμμετρικής κρυπτογραφίας, όπου E_{sym} η συνάρτηση κρυπτογράφησης και D_{sym} η συνάρτηση αποκρυπτογράφησης. Παράδειγμα τέτοιου αλγορίθμου είναι ο AES (Ενότητα 1.3).
- Να επιλεγούν οι παράμετροι τομέα της ελλειπτικής καμπύλης: Πιο συγκεκριμένα, να καθορισθούν οι παράμετροι (p, a, b, G, n, h) για μια καμπύλη πάνω σε πρώτα πεπερασμένα σώματα $GF(p)$ ή οι παράμετροι $(m, f(x), a, b, G, n, h)$ για μια καμπύλη πάνω σε δυαδικά πεπερασμένα σώματα $GF(2^m)$.
- Να δημιουργηθεί ένα ζεύγος κλειδιών: Το ιδιωτικό κλειδί s_k επιλέγεται τυχαία από το σύνολο $\{1, \dots, n-1\}$ και το δημόσιο κλειδί υπολογίζεται ως $P_k = s_k \cdot G$.
- Να οριστούν προαιρετικά οι κοινόχρηστες πληροφορίες S_1 και S_2 .
- Το Ο αποτελεί σημείο στο άπειρο.

2.6.2.2 Αλγόριθμος Κρυπτογράφησης

Για να κρυπτογραφηθεί ένα μήνυμα m , ο αλγόριθμος που ακολουθείται είναι ο εξής:

1. Επιλέγουμε έναν τυχαίο αριθμό $r \in [1, n - 1]$ και υπολογίζουμε το σημείο $R = r \cdot G$.
2. Παράγουμε ένα κοινό μυστικό $S = P_x$, όπου $P = (P_x, P_y) = r \cdot P_k$ (και $P \neq O$).
3. Χρησιμοποιούμε την συνάρτηση KDF για να παράγουμε τα συμμετρικά κλειδιά κρυπτογράφησης και τα κλειδιά MAC: $k_E \parallel k_{MAC} = \text{KDF}(S \parallel S_1)$.
4. Κρυπτογραφούμε το μήνυμα m με βάση το συμμετρικό σύστημα κρυπτογράφησης ως εξής: $c = E_{sym}(k_E, m)$.
5. Υπολογίζουμε την ετικέτα (tag) του κρυπτογραφημένου μηνύματος και του S_2 : $d = \text{MAC}(k_{MAC}, c \parallel S_2)$
6. Το αποτέλεσμα της κρυπτογράφησης θα είναι: $R \parallel c \parallel d$.

2.6.2.3 Αλγόριθμος Αποκρυπτογράφησης

Για να αποκρυπτογραφηθεί το κρυπτοκείμενο $R \parallel c \parallel d$, ο αλγόριθμός που ακολουθείται είναι ο εξής:

1. Παράγουμε το κοινό μυστικό: $S = P_x$, όπου $P = (P_x, P_y) = s_k \cdot R$ (είναι το ίδιο με αυτό που προέρχεται με χρήση του δημόσιου κλειδιού, επειδή $P = s_k \cdot R = s_k \cdot r \cdot G = r \cdot s_k \cdot G = r \cdot P_k$), ή η αποτυγχάνει εάν $P = O$.
2. Παράγουμε τα συμμετρικά κλειδιά κρυπτογράφησης και τα κλειδιά MAC με τον ίδιο τρόπο: $k_E \parallel k_{MAC} = \text{KDF}(S \parallel S_1)$.
3. Χρησιμοποιούμε το κώδικα αυθεντικοποίησης μηνύματος (MAC) για να ελέγξουμε την εγκυρότητα της ετικέτας (tag) και αποτυγχάνει εάν $d \neq \text{MAC}(k_{MAC}, c \parallel S_2)$.
4. Χρησιμοποιούμε το συμμετρικό σύστημα κρυπτογράφησης για να αποκρυπτογραφήσουμε το μήνυμα $m = D_{sym}(k_E, c)$.

2.6.3 Αλγόριθμος Ψηφιακών Υπογραφών ECDSA

Ο αλγόριθμος ψηφιακών υπογραφών ECDSA (Elliptic Curve Digital Signature Algorithm) αποτελεί μια παραλλαγή του αλγορίθμου ψηφιακών υπογραφών DSA (Ενότητα 2.5) με χρήση ελλειπτικών καμπυλών, αποτελώντας και αυτός πρότυπο ψηφιακών υπογραφών του FIPS (Federal Information Processing Standards) [22]. Τόσο το πρότυπο FIPS 186-4 [22] όσο και το ANSI X9.62 [36] ορίζουν ως ελάχιστο μέγεθος κλειδιού τα 1024 bits για τον RSA και τον DSA, ενώ μόλις 160 bits για ECC, θεωρώντας ότι παρέχουν ισοδύναμη ασφάλεια με συμμετρική κρυπτογραφία με μέγεθος κλειδιού 80 bits (βλέπε Πίνακα 2.1). Συγκριτικά, τα μηνύματα που υπογράφονται με ένα κλειδί RSA 1024 bits παράγουν μια ψηφιακή υπογραφή 128 bytes, ενώ το ίδιο μήνυμα υπογεγραμμένο με ένα κλειδί ECDSA 192 bits δημιουργεί μια ψηφιακή υπογραφή 48 bytes.

2.6.3.1 Προαπαιτούμενα Δημιουργίας Υπογραφής

Για να δημιουργηθεί μια ψηφιακή υπογραφή ECDSA, προαπαιτούνται τα ακόλουθα:

- Να επιλεγεί μια ασφαλής συνάρτηση σύνοψης H με έξοδο μήκους $|H|$ bit, τα οποία μετασχηματίζονται με την μορφή ενός ακέραιου αριθμού. Παράδειγμα αποτελεί η οικογένεια συναρτήσεων σύνοψης SHA-2 (βλέπε Κεφάλαιο 3).
- Να επιλεγούν οι παράμετροι τομέα της ελλειπτικής καμπύλης: Πιο συγκεκριμένα, να καθορισθούν οι παράμετροι (p, a, b, G, n, h) για μια καμπύλη πάνω σε πρώτα πεπερασμένα σώματα $GF(p)$ ή οι παράμετροι $(m, f(x), a, b, G, n, h)$ για μια καμπύλη πάνω σε δυαδικά πεπερασμένα σώματα $GF(2^m)$.
- Να δημιουργηθεί ένα ζεύγος κλειδιών: Το ιδιωτικό κλειδί s_k επιλέγεται τυχαία από το σύνολο $\{1, \dots, n-1\}$ και το δημόσιο κλειδί υπολογίζεται ως $P_k = s_k \cdot G$.

2.6.3.2 Δημιουργία Υπογραφής

Η διαδικασία που ακολουθείται από τον υπογράφοντα για την δημιουργία υπογραφής ECDSA ενός μηνύματος m είναι η ακόλουθη:

1. Υπολογίζει την σύνοψη του μηνύματος m με βάση την επιλεγέσια συνάρτηση σύνοψης $e = H(m)$.
2. Έστω z είναι τα L_n πιο αριστερά bits του e , όπου L_n είναι το πλήθος των bits της τάξης ομάδας n .
3. Επιλέγει έναν τυχαίο ακέραιο αριθμό k από το σύνολο $\{1, \dots, n-1\}$. Το τυχαία επιλεγμένο k πρέπει να παραμείνει κρυφό και να είναι διαφορετικό για διαφορετικές υπογραφές.
4. Υπολογίζει το σημείο καμπύλης $(x_1, y_1) = k \cdot G$.
5. Υπολογίζει το $r = x_1 \mod n$. Εάν το $r = 0$ επιστρέφουμε στο βήμα 3.
6. Υπολογίζει το $s = k^{-1}(z + r \cdot s_k) \mod n$. Εάν το $s = 0$ επιστρέφουμε στο βήμα 3.
7. Η ψηφιακή υπογραφή αποτελείται από το ζεύγος $S_{s_k}(m, k) = (r, s)$. Αξίζει να σημειωθεί ότι και το $(r, -s \mod n)$ αποτελεί επίσης έγκυρη υπογραφή.

2.6.3.3 Επαλήθευση Υπογραφής

Αντίστοιχα, η διαδικασία που ακολουθείται από τον παραλήπτη για την επαλήθευση υπογραφής ECDSA ενός μηνύματος m είναι η εξής:

1. Το δημόσιο κλειδί P_k του υπογράφοντα είναι δημόσια διαθέσιμο και βρίσκεται στην πλευρά του παραλήπτη.
2. Επαληθεύει ότι το P_k δεν είναι ίσο με το O και ότι οι συντεταγμένες του είναι έγκυρες.
3. Επαληθεύει ότι το P_k βρίσκεται στην καμπύλη.
4. Επαληθεύει ότι $n \cdot P_k = O$.
5. Ελέγχει ότι τα r και s είναι ακέραιοι αριθμοί και ανήκουν στο σύνολο $\{1, \dots, n-1\}$. Εάν όχι, τότε η υπογραφή δεν είναι έγκυρη.
6. Υπολογίζει το $e = H(m)$, όπου H είναι η ίδια συνάρτηση σύνοψης που χρησιμοποιείται στη δημιουργία της υπογραφής.

7. Έστω z είναι τα L_n πιο αριστερά bits του e .
8. Υπολογίζει τα $u_1 = z \cdot s^{-1} \pmod n$ και $u_2 = r \cdot s^{-1} \pmod n$.
9. Υπολογίζει το σημείο καμπύλης $(x_1, y_1) = u_1 \cdot G + u_2 \cdot P_k$. Εάν $(x_1, y_1) = O$, τότε η υπογραφή δεν είναι έγκυρη.
10. Η ψηφιακή υπογραφή είναι έγκυρη, εάν και μόνο εάν $V_{P_k}(m, (r, s)) = x_1 \pmod n \equiv r$.

2.7 Ασκήσεις - Εργασίες

Ασκήσεις

2.7.1 Δεδομένου ότι έχουμε το κρυπτοσύστημα RSA:

- (1) Εάν η Αλίκη επιλέξει δύο πρώτους αριθμούς $p = 5$ και $q = 11$, τότε να υπολογίσετε το n και το $\Phi(n)$.
- (2) Στην συνέχεια η Αλίκη επιλέγει το $e = 3$ ως το δημόσιο κλειδί της. Είναι αυτή η επιλογή της σωστή; Με βάση αυτό το e να υπολογίσετε το d που αποτελεί το ιδιωτικό κλειδί της.
- (3) Τώρα χρειάζεται να στείλετε ένα μήνυμα $m = 4$ στην Αλίκη. Κρυπτογραφήστε το μήνυμα m με βάση τον δημόσιο εκθέτη της Αλίκης. Ποιο είναι το κρυπτοκείμενο c ;
- (4) Τέλος, η Αλίκη λαμβάνει το κρυπτοκείμενο c . Επαληθεύστε εάν μπορεί με την διαδικασία της αποκρυπτογράφησης να ανακτήσει το αρχικό μήνυμα m .

2.7.2 Θέλουμε να υπογράψουμε ψηφιακά ένα μήνυμα $m = 65$ με το κρυπτοσύστημα RSA, και δεδομένου ότι ανήκουν στην Αλίκη το $n = p \cdot q = 221$ και το $e = 13$ (και είναι δημόσια διαθέσιμα), να επαληθεύσετε εάν η υπογραφή $s = 182$ είναι έγκυρη.

2.7.3 Η Αλίκη αρχικοποιεί τις παραμέτρους ενός κρυπτοσυστήματος RSA και για κακή της τύχη ένας κρυπταναλυτής μαθαίνει ότι $n = 493$ και $\Phi(n) = 448$, τότε:

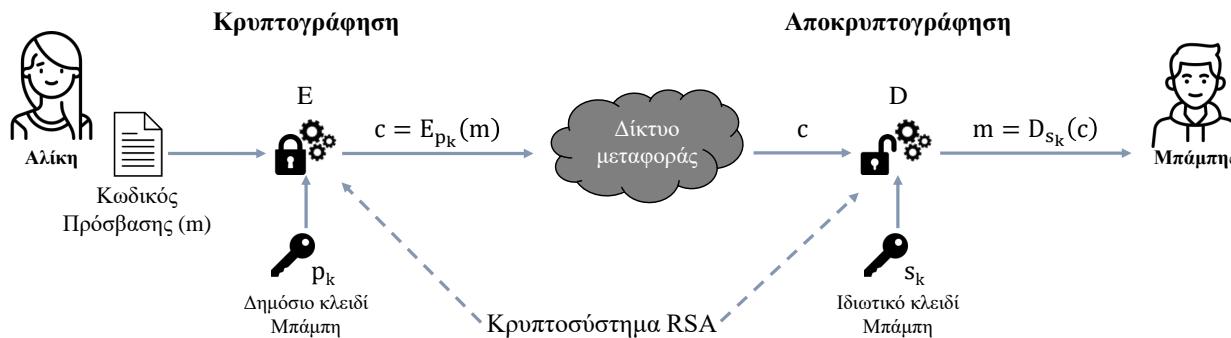
- (1) Βρείτε τους δύο παράγοντες p και q που συνθέτουν το n (γνωστό ως πρόβλημα παραγοντοποίησης).
- (2) Αν υποθέσουμε ότι το δημόσιο κλειδί της Αλίκης είναι $e = 3$, να βρείτε το ιδιωτικό κλειδί d της Αλίκης.

2.7.4 Ο Μπάμπης έχει δημοσιεύσει το δημόσιο του κλειδί $(p, g, y) = (283, 60, 216)$ για το κρυπτοσύστημα ElGamal, και το αντίστοιχο ιδιωτικό του κλειδί που κρατά κρυφό είναι το $a = 7$. Η Αλίκη στέλνει το κρυπτογραφημένο μήνυμα $(\gamma, \delta) = (78, 218)$ στον Μπάμπη. Ποιο είναι το αποκρυπτογραφημένο μήνυμα m ;

2.7.5 Ο Μπάμπης έχει δημοσιεύσει το δημόσιο του κλειδί $(n, g) = (221, 4886)$ για το κρυπτοσύστημα Pailier. Η Αλίκη θέλει να στείλει με ασφάλεια το μήνυμα $m = 123$ στον Μπάμπη κάνοντας χρήση του τυχαίου αριθμού $r = 48$. Ποιο είναι το κρυπτογραφημένο μήνυμα c που θα στείλει η Αλίκη;

Εργασίες

- 2.7.1** Σε αυτήν την εργασία θα εξετάσετε την περίπτωση όπου 2 χρήστες που βρίσκονται σε διαφορετικές γεωγραφικές τοποθεσίες θέλουν ο ένας να στείλει στον άλλο έναν κωδικό πρόσβασης για μια υπηρεσία (π.χ. μιας ιστοσελίδας) εξασφαλίζοντας με αυτόν τον τρόπο την εμπιστευτικότητα της επικοινωνίας. Συγκεκριμένα, θα χρησιμοποιήσετε κρυπτογραφία δημοσίου κλειδιού για την κρυπτογράφηση του κωδικού πρόσβασης από τον αποστολέα με βάση το δημόσιο κλειδί του παραλήπτη με σκοπό την ασφαλή μετάδοσή του. Αντίστοιχα, ο παραλήπτης θα πρέπει να μπορεί να αποκρυπτογράφησε τον κωδικό πρόσβασης με χρήση του ιδιωτικού του κλειδιού. Στην εργασία αυτή να χρησιμοποιήσετε το κρυπτοσύστημα δημοσίου κλειδιού RSA κάνοντας χρήση του εργαλείου [CrypTool 2](#). Η λειτουργικότητα που καλείστε να υλοποιήσετε απεικονίζεται στο Σχήμα 2.3.



Σχήμα 2.3: Διαδικασία κρυπτογράφησης και αποκρυπτογράφησης του κωδικού πρόσβασης με βάση το κρυπτοσύστημα RSA.

- 2.7.2** Σε αυτήν την εργασία θα δοκιμάσετε διάφορα κρυπτοσυστήματα δημοσίου κλειδιού με χρήση της γλώσσας προγραμματισμού [Java](#) και το περιβάλλον ανάπτυξης [Eclipse IDE for Java Developers](#). Πιο αναλυτικά, θα δοκιμάσετε τους αλγορίθμους δημιουργίας κλειδιών, κρυπτογράφησης, αποκρυπτογράφησης, δημιουργίας/επαλήθευσης υπογραφών για τα κρυπτοσυστήματα RSA, ElGamal, και Paillier. Κάνοντας χρήση του Eclipse Project “[crypto_chap02](#)”, οι δοκιμές που θα πρέπει να γίνουν είναι οι εξής:



- (1) Για να δοκιμάσετε την κρυπτογράφηση/αποκρυπτογράφηση του κρυπτοσυστήματος RSA εκτελέστε το αρχείο `TestRSAEncr.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να κρυπτογραφήσετε το αριθμό -1 και δείτε ποιο είναι το αποτέλεσμα αποκρυπτογράφησης (γιατί δεν είναι -1). Παρατηρήστε ότι στην περίπτωση του padded-RSA η κρυπτογράφηση του ίδιου αριθμού έχει ως αποτέλεσμα διαφορετικό κρυπτοκείμενο κάθε φορά σε αντίθεση με την βασική υλοποίηση του RSA που είναι πάντα το ίδιο κρυπτοκείμενο.
- (2) Για να δοκιμάσετε την κρυπτογράφηση/αποκρυπτογράφηση του κρυπτοσυστήματος ElGamal εκτελέστε το αρχείο `TestElgamalEncr.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να δημιουργήσετε ένα ζευγάρι κλειδιών χωρίς να κάνετε χρήση των προϋπολογισμένων τιμών p και g , δηλαδή να κάνετε χρήση της γραμμής “`ElgamalKeyPair pkp = new ElgamalKeyPair(512);`” (γιατί αργεί πλέον πολύ;).
- (3) Για να δοκιμάσετε την κρυπτογράφηση/αποκρυπτογράφηση του κρυπτοσυστήματος Paillier εκτελέστε το αρχείο `TestPaillierEncr.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να πολλαπλασιάσετε τα δύο κρυπτοκείμενα μεταξύ τους πριν την αποκρυπτογράφηση (δηλ. “`em2 = em2.multiply(em1);`”), ποιο είναι το αποτέλεσμα της αποκρυπτογράφησης;

- (4) Για να δοκιμάσετε την δημιουργία/επαλήθευση υπογραφών του κρυπτοσυστήματος RSA εκτελέστε το αρχείο `TestRSASign.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε πριν την επαλήθευση των υπογραφών να αλλάξετε τιμή σε ένα από τα δύο μηνύματα (π.χ. "`m2 = BigInteger.valueOf(4);`"), η υπογραφή συνεχίζει να είναι έγκυρη;
- (5) Για να δοκιμάσετε την δημιουργία/επαλήθευση υπογραφών του κρυπτοσυστήματος ElGamal εκτελέστε το αρχείο `TestElgamalSign.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να αντικαταστήσετε ένα από τα δύο μηνύματα βάζοντας κάποια φράση (π.χ. "`BigInteger m2 = new BigInteger("Crypto".getBytes());`"), η υπογραφή να είναι έγκυρη; Εάν τυπώσετε το μήνυμα αυτό (π.χ. "`System.out.println("m2: "+m2);`") συνεχίζει να είναι κείμενο; Γιατί όχι;

Βιβλιογραφία

- [1] Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 2018.
- [2] George Drosatos. "Algorithms for Exploiting Personal Location Data While Protecting Privacy: Application in Provision of Medical Services". Greek. MA thesis. University Campus, Xanthi 67100, Greece: Department of Electrical and Computer Engineering, Democritus University of Thrace, Mar. 2010.
- [3] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. 20th Anniversary. John Wiley & Sons, 2015. ISBN: 978-1-119-09672-6.
- [4] Ronald L Rivest, Adi Shamir, and Leonard Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Communications of the ACM* 21.2 (Feb. 1978), pp. 120–126. ISSN: 0001-0782. doi: 10.1145/359340.359342.
- [5] William Buchanan and Alan Woodward. "Will quantum computers be the end of public key encryption?" In: *Journal of Cyber Security Technology* 1.1 (2017), pp. 1–22. doi: 10.1080/23742917.2016.1226650.
- [6] Dan Boneh. "Twenty Years of Attacks on the RSA Cryptosystem". In: *Notices of the American Mathematical Society* 46.2 (1999), pp. 203–213. ISSN: 0002-9920.
- [7] Johannes Blömer, Martin Otto, and Jean-Pierre Seifert. "A New CRT-RSA Algorithm Secure against Bellcore Attacks". In: *Proceedings of the 10th ACM Conference on Computer and Communications Security*. CCS '03. Washington D.C., USA: ACM, 2003, pp. 311–320. ISBN: 1581137389. doi: 10.1145/948109.948151.
- [8] Derek Atkins, Michael Graff, Arjen K. Lenstra, and Paul C. Leyland. "The Magic Words are Squeamish Ossifrage". In: *Advances in Cryptology — ASIACRYPT'94*. Ed. by Josef Pieprzyk and Reihana Safavi-Naini. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 261–277. ISBN: 978-3-540-49236-8. doi: 10.1007/BFb0000440.
- [9] F. Bahr, M. Boehm, Jens Franke, and Thorsten Kleinjung. *We have factored RSA200 by GNFS*. <https://members.loria.fr/PZimmermann/records/rsa200>. 2005.
- [10] Fabrice Boudot et al. *Factorization of RSA-250*. <https://lists.gforge.inria.fr/pipermail/cado-nfs-discuss/2020-February/001166.html>. 2020.
- [11] Shafi Goldwasser and Silvio Micali. "Probabilistic Encryption". In: *Journal of Computer and System Sciences* 28.2 (1984), pp. 270–299. ISSN: 0022-0000. doi: 10.1016/0022-0000(84)90070-9.

- [12] Kathleen Moriarty, Burt Kaliski, Jakob Jonsson, and Andreas Rusch. *PKCS #1: RSA Cryptography Specifications Version 2.2*. RFC 8017. <https://www.rfc-editor.org/rfc/rfc8017.txt>. RFC Editor, Nov. 2016.
- [13] Whitfield Diffie and Martin Hellman. “New Directions in Cryptography”. In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. doi: 10.1109/TIT.1976.1055638.
- [14] Taher Elgamal. “A public key cryptosystem and a signature scheme based on discrete logarithms”. In: *IEEE Transactions on Information Theory* 31.4 (1985), pp. 469–472. doi: 10.1109/TIT.1985.1057074.
- [15] David Pointcheval and Jacques Stern. “Security Arguments for Digital Signatures and Blind Signatures”. In: *Journal of Cryptology* 13.3 (2000), pp. 361–396. doi: 10.1007/s001450010003.
- [16] George Drosatos. “Utilization and Protection of Personal Data in Ubiquitous Computing Environments”. English. PhD thesis. University Campus, Xanthi 67100, Greece: Department of Electrical and Computer Engineering, Democritus University of Thrace, July 2013. doi: 10.12681/eadd/30085.
- [17] Pascal Paillier. “Public-Key Cryptosystems Based on Composite Degree Residuosity Classes”. In: *Advances in Cryptology — EUROCRYPT ’99*. Ed. by Jacques Stern. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 223–238. ISBN: 978-3-540-48910-8.
- [18] Olivier Baudron, Pierre-Alain Fouque, David Pointcheval, Jacques Stern, and Guillaume Poupard. “Practical Multi-Candidate Election System”. In: *Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing*. PODC ’01. Newport, Rhode Island, USA: Association for Computing Machinery, 2001, pp. 274–283. ISBN: 1581133839. doi: 10.1145/383962.384044.
- [19] Ivan Damgård and Mads Jurik. “A Generalisation, a Simplification and Some Applications of Paillier’s Probabilistic Public-Key System”. In: *Public Key Cryptography*. Ed. by Kwangjo Kim. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 119–136. ISBN: 978-3-540-44586-9. doi: 10.1007/3-540-44586-2_9.
- [20] Pascal Paillier and David Pointcheval. “Efficient Public-Key Cryptosystems Provably Secure Against Active Adversaries”. In: *Advances in Cryptology – ASIACRYPT’99*. Ed. by Kwok-Yan Lam, Eiji Okamoto, and Chaoping Xing. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 165–179. ISBN: 978-3-540-48000-6. doi: 10.1007/978-3-540-48000-6_14.
- [21] Ronald H. Brown and Arati Prabhakar. “Digital signature standard (DSS)”. In: *Federal Information Processing Standards Publication FIPS PUB 186* (1994).
- [22] Cameron F. Kerry and Patrick D. Gallagher. “Digital signature standard (DSS)”. In: *Federal Information Processing Standards Publication FIPS PUB 186-4* (2013), pp. 1–121.
- [23] Elaine Barker, William Barker, William Burr, William Polk, and Miles Smid. “NIST Special Publication 800-57”. In: *National Institute of Standards and Technology (NIST) 800.57* (2007), pp. 1–142.
- [24] Daniel R. L. Brown. *Standards for Efficient Cryptography 1 (SEC 1): Elliptic Curve Cryptography*. 2nd ed. <https://www.secg.org/sec1-v2.pdf>. Certicom Research, May 2009, pp. 1–138.
- [25] Víctor Gayoso Martínez, Luis Hernández Encinas, and Carmen Sánchez Ávila. “A Survey of the Elliptic Curve Integrated Encryption Scheme”. In: *Journal of Computer Science and Engineering* 2.2 (2010), pp. 7–13.
- [26] Neal Koblitz. “Elliptic Curve Cryptosystems”. In: *Mathematics of Computation* 48.177 (1987), pp. 203–209. doi: 10.1090/S0025-5718-1987-0866109-5.

- [27] Victor S. Miller. "Use of Elliptic Curves in Cryptography". In: *Advances in Cryptology — CRYPTO '85*. Ed. by Hugh C. Williams. Berlin, Heidelberg: Springer Berlin Heidelberg, 1986, pp. 417–426. ISBN: 978-3-540-39799-1. doi: 10.1007/3-540-39799-X_31.
- [28] Darrel Hankerson, Alfred Menezes, and Scott Vanstone. *Guide to Elliptic Curve Cryptography*. Springer Science + Business Media, 2004. ISBN: 0-387-95273-X.
- [29] Joseph H. Silverman. *The Arithmetic of Elliptic Curves*. 2nd ed. Springer, 2009. ISBN: 978-0-387-09493-9. doi: 10.1007/978-0-387-09494-6.
- [30] Daniel R. L. Brown. *Standards for Efficient Cryptography 2 (SEC 2): Recommended Elliptic Curve Domain Parameters*. 2nd ed. <http://www.secg.org/sec2-v2.pdf>. Certicom Research, Jan. 2010, pp. 1–33.
- [31] Manfred Lochter and Johannes Merkle. *Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation*. RFC 5639. <https://www.rfc-editor.org/rfc/rfc5639.txt>. RFC Editor, Mar. 2010.
- [32] Michel Abdalla, Mihir Bellare, and Phillip Rogaway. *DHAES: An Encryption Scheme Based on the Diffie-Hellman Problem*. Cryptology ePrint Archive, Report 1999/007. <https://ia.cr/1999/007>. 1999.
- [33] *Public Key Cryptography for the Financial Services Industry Key Agreement and Key Transport Using Elliptic Curve Cryptography*. Standard. American National Standards Institute, Nov. 2001, pp. 1–415.
- [34] *Standard Specifications for Public-Key Cryptography - Amendment 1: Additional Techniques*. Standard. Institute of Electrical and Electronics Engineers, Sept. 2004, pp. 1–167. doi: 10.1109/IEEESTD.2004.94612.
- [35] *Information technology – Security techniques – Encryption algorithms – Part 2: Asymmetric ciphers*. Standard. International Organization for Standardization/International Electrotechnical Commission, May 2006, pp. 1–125.
- [36] *Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*. Standard. American National Standards Institute, Nov. 2005, pp. 1–163.

ΚΕΦΑΛΑΙΟ 3

ΣΥΝΑΡΤΗΣΕΙΣ ΣΥΝΟΨΗΣ

Περίληψη

Οι συναρτήσεις σύνοψης [1] μετατρέπουν την είσοδο δεδομένων αυθαίρετου μεγέθους σε έξοδο σταθερού μικρού μεγέθους (συμπίεση), ανάλογα με τη συνάρτηση σύνοψης που χρησιμοποιείται. Λόγω των ιδιοτήτων τους, με έμφαση αυτών της μονόδρομης λειτουργίας αλλά και της ανθεκτικότητας σε συγκρούσεις, βρίσκουν πολλές εφαρμογές στην ασφάλεια πληροφοριών. Παραδείγματα εφαρμογών τους είναι οι ψηφιακές υπογραφές, ο έλεγχος ακεραιότητας μηνυμάτων και αρχείων, ο διαμοιρασμός αποτυπωμάτων κυβερνοαπειλών και οι αλυσίδες μπλοκ. Στο κεφάλαιο αυτό αναλύονται τα βασικά χαρακτηριστικά και οι ιδιότητες των συναρτήσεων σύνοψης και παρουσιάζονται ο τρόπος λειτουργίας βασικών συναρτήσεων, όπως οι SHA-2, SHA-3, Whirlpool, και BLAKE. Πιο αναλυτικά, στην Ενότητα 3.1 γίνεται μια προσπάθεια ορισμού των συναρτήσεων σύνοψης και των ιδιοτήτων τους, ενώ στην Ενότητα 3.2 παρουσιάζονται διάφορα δομικά στοιχεία γνωστών συναρτήσεων σύνοψης. Στις Ενότητες 3.3 και 3.4 γίνεται περιγραφή των συναρτήσεων σύνοψης MD5 και Whirlpool, αντίστοιχα. Επιπρόσθετα, στις Ενότητες 3.5 και 3.6 γίνεται αναλυτική παρουσίαση των οικογενειών συναρτήσεων σύνοψης SHA και BLAKE, αντίστοιχα, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: –.

3.1 Ορισμός των Συναρτήσεων Σύνοψης

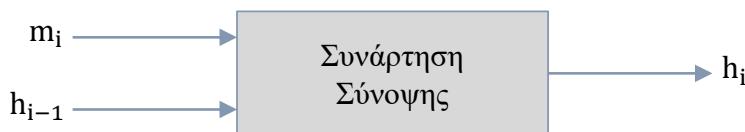
Μια συνάρτηση σύνοψης (ή συνάρτηση κατακερματισμού) (Hash Function) [1] είναι μια συνάρτηση που δέχεται ως είσοδο δεδομένα αυθαίρετου μήκους (συχνά αποκαλούνται «μήνυμα») και παράγει ως έξοδο μια συστοιχία από bits σταθερού μεγέθους, γνωστή ως σύνοψη (message digest ή hash value) ή αποτύπωμα (fingerprint). Χαρακτηρίζεται από ένα μονόδρομο τρόπο λειτουργίας (one-way function), δηλαδή είναι πρακτικά αδύνατον να ανακτηθεί από την έξοδο της συνάρτησης η αρχική είσοδός της. Επομένως, μπορεί να θεωρηθεί ως κάτι το αντίθετο από μια γεννήτρια ψευδοτυχαίων αριθμών (βλέπε Κεφάλαιο 4), η οποία παράγει με βάση μια συμβολοσειρά μικρού και σταθερού μήκους, μια αυθαίρετα μεγάλη συμβολοσειρά. Στην πραγματι-

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

κότητα, οι μονόδρομες συναρτήσεις σύνοψης στηρίζονται στην ιδέα μιας συνάρτησης συμπίεσης (compression function). Αποτελέσματα αυτής της μονόδρομης συνάρτησης είναι μια σύνοψη (hash) με μήκος n_h , για δεδομένη είσοδο μήκους n_m . Οι είσοδοι της συνάρτησης συμπίεσης F είναι ένα μπλοκ δεδομένων (m_i) του μηνύματος m και η έξοδος σύνοψης (h_{i-1}) του προηγούμενου μπλοκ δεδομένων (m_{i-1}) [2, 3] (Σχήμα 3.1). Η έξοδος της συνάρτησης σύνοψης H είναι ουσιαστικά η σύνοψη όλων των μπλοκ μέχρι εκείνο το σημείο. Δηλαδή, η σύνοψη (hash) ολόκληρου του του μηνύματος m θα δίνεται από την ακόλουθη εξίσωση:

$$H(m) = \begin{cases} h_i = F(m_i, IV) & \text{για } i = 0, \text{ όπου } IV \text{ μια αρχική τιμή (διάνυσμα αρχικοποίησης) της σύνοψης} \\ h_i = F(m_i, h_{i-1}) & \text{για } i > 0 \end{cases}$$



Σχήμα 3.1: Συνάρτηση σύνοψης.

Ιδανικά μια συνάρτηση σύνοψης θα πρέπει να διαθέτει τις ακόλουθες κύριες ιδιότητες (οι αντίστοιχες ιδιότητες ασφαλείας παρατίθενται στον Ορισμό 3.1):

- Να είναι ντετερμινιστική, δηλαδή για το ίδιο μήνυμα το αποτέλεσμα της συνάρτησης να είναι πάντα η ίδια σύνοψη.
- Να είναι γρήγορη στον υπολογισμό της σύνοψης για κάθε μήνυμα.
- Να είναι αδύνατον να βρεθεί ένα μήνυμα που να παράγει μια επιθυμητή σύνοψη.
- Να είναι αδύνατον να βρεθούν δύο διαφορετικά μηνύματα με την ίδια σύνοψη.
- Με μια και μόνο μικρή αλλαγή στο μήνυμα θα πρέπει να αλλάζει η σύνοψη τόσο εκτεταμένα ώστε να μην μπορεί να συσχετιστεί με την προηγούμενη.

Αν και αρχικά ο στόχος των συναρτήσεων σύνοψης ήταν να παρέχουν μόνο ακεραιότητα (integrity) και αυθεντικότητα (authenticity), έμμεσα συμβάλλουν στην εξασφάλιση της εμπιστευτικότητας (confidentiality) και της διαθεσιμότητας (availability) των δεδομένων. Επιπλέον, οι συναρτήσεις σύνοψης αποτελούν βασικά δομικά στοιχεία διαφόρων σχημάτων κρυπτογράφησης (π.χ. ο padded-RSA (PKCS #1 v2.2) [4]) και χρησιμοποιούνται επίσης για πιο αποτελεσματική αποθήκευση και ανάκτηση δεδομένων (π.χ. σε υπηρεσίες αποθήκευσης νέφους (cloud) κάνοντας χρήση αποδείξεων αποθήκευσης (Proofs of Storage – PoS) [5] και αποθετηρίων ζευγών κλειδιού-τιμής (Key-Value Stores) [6]).

Ορισμός 3.1 (Ιδιότητες Ασφαλείας των Συναρτήσεων Σύνοψης). Οι επιθυμητές ιδιότητες ασφαλείας που καλούνται να ικανοποιούν οι συναρτήσεις σύνοψης είναι οι εξής:

- (1) Αντίσταση πρώτου ορίσματος (pre-image resistance): Δεδομένης μιας τιμής σύνοψης h , θα πρέπει να είναι υπολογιστικά αδύνατο να βρεθεί κάποιο μήνυμα m τέτοιο ώστε $h = H(m)$.
- (2) Αντίσταση δεύτερου ορίσματος (second pre-image resistance): Δεδομένου ενός μηνύματος m_1 , θα πρέπει να είναι υπολογιστικά αδύνατο να βρεθεί ένα διαφορετικό μήνυμα m_2 τέτοιο ώστε $H(m_1) = H(m_2)$.
- (3) Δυσκολία εύρεσης συγκρούσεων (collision resistance): Λαμβάνοντας υπόψη δύο μηνύματα m_1 και m_2 , θα πρέπει να είναι υπολογιστικά αδύνατο να βρεθεί μια τιμή σύνοψη h τέτοια ώστε $h = H(m_1) = H(m_2)$.

Σειρά ισχύος (υπό προϋποθέσεις): (3) \Rightarrow (2) \Rightarrow (1).

3.2 Δομικά Στοιχεία των Συναρτήσεων Σύνοψης

Όλες οι συναρτήσεις σύνοψης γενικού σκοπού χωρίζουν τα δεδομένα εισόδου σε μπλοκ σταθερού μήκους και τα επεξεργάζονται χρησιμοποιώντας μια συνάρτηση συμπίεσης (compression function). Η συνάρτηση συμπίεσης λαμβάνει ως είσοδο δεδομένα σταθερού μήκους και παράγει έξοδο μικρότερου αλλά σταθερού μήκους, επίσης. Ο συνδυασμός κλήσεων μιας συνάρτησης συμπίεσης για την επεξεργασία εισόδου (μηνύματος) αυθαίρετου μήκους ονομάζεται λειτουργία επανάληψης (iteration mode). Σε αυτή την ενότητα θα παρουσιάσουμε την λειτουργία επανάληψης που χρησιμοποιείται στις συναρτήσεις σύνοψης MD5, SHA-1 και SHA-2 (η λεγόμενη κατασκευή Merkle-Damgård) καθώς και τις λειτουργίες επανάληψης που χρησιμοποιούνται από τις πιο πρόσφατες συναρτήσεις σύνοψης, όπως οι SHA-3 και BLAKE3. Τέλος, θα αναφερθούμε σε συναρτήσεις συμπίεσης που κάνουν χρήση κρυπτογράφησης μπλοκ, όπως στην συνάρτηση σύνοψης Whirlpool.

3.2.1 Κατασκευή Merkle-Damgård

Η κατασκευή Merkle-Damgård [7] υλοποιεί μια λειτουργία επανάληψης που αποτελείται από δύο βήματα: (1) την πλήρωση των δεδομένων (padding), και (2) την συμπίεση των δεδομένων (compression). Παρακάτω περιγράφονται αναλυτικά αυτά τα δύο βήματα.

3.2.1.1 Πλήρωση Δεδομένων

Τα δεδομένα εισόδου μπορούν να έχουν αυθαίρετο μέγεθος (πλήθος από bit). Ωστόσο, η λειτουργία επανάληψης επεξεργάζεται μπλοκ δεδομένων μεγέθους m bit. Είναι επομένως απαραίτητο να μετασχηματιστούν τα δεδομένα εισόδου σε μια ακολουθία μπλοκ μεγέθους m bit το καθένα με μια αναστρέψιμη διαδικασία, ώστε να αποφευχθεί κάποια σύγκρουση (collision). Με άλλα λόγια, τα αρχικά δεδομένα να μπορούν να καθορίζονται μοναδικά λαμβάνοντας υπόψη τα δεδομένα μετά την πλήρωση (padding).

Για παράδειγμα, στις συναρτήσεις σύνοψης SHA-1, SHA-224 και SHA-256, το μέγεθος του μπλοκ είναι $m = 512$ bits [7]. Η πλήρωση με δεδομένα μεγέθους l bit γίνεται ως εξής:

1. Προσαρτάται ένα bit με τιμή “1” στο τελευταίο μπλοκ.
2. Στη συνέχεια, προσαρτώνται k bit με τιμή “0”, όπου $k \geq 0$ και αποτελεί την μικρότερη λύση της εξίσωσης $l + 1 + k \equiv 448 \pmod{512}$.
3. Τέλος, προσαρτάται η τιμή (μορφής unsigned big-endian μεγέθους 64 bit) του συνολικού μήκους του μηνύματος μεγέθους l bit πριν την πλήρωσή του.

Αυτή η διαδικασία εγγυάται ότι το μέγεθος των δεδομένων μετά την πλήρωση είναι πολλαπλάσιο του 512.

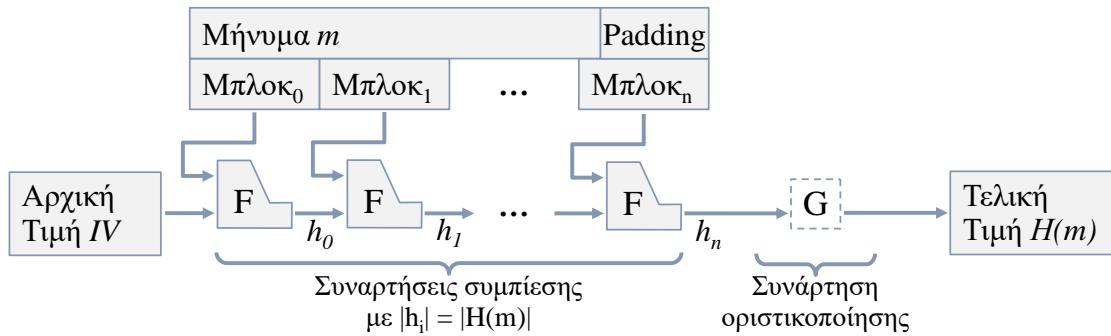
Αντίστοιχα, στις συναρτήσεις σύνοψης SHA-384 και SHA-512, όπου το μέγεθος του μπλοκ είναι $m = 1024$ bit, η πλήρωση είναι παρόμοια, εκτός από το ότι το k θα πρέπει να ικανοποιεί την εξίσωση $l + 1 + k \equiv 896 \pmod{1024}$ και ότι το l αναπαριστάται με 128 bits [7].

3.2.1.2 Συμπίεση Δεδομένων

Μετά την πλήρωση δεδομένων, η συνάρτηση σύνοψης επεξεργάζεται μια ακολουθία από μπλοκ m_0, \dots, m_n με επαναληπτική εφαρμογή μιας συνάρτησης συμπίεσης ως εξής:

$$h_i = \begin{cases} F(m_i, IV) & \text{για } i = 0, \text{ όπου } IV \text{ μια προκαθορισμένη αρχική τιμή} \\ F(m_i, h_{i-1}) & \text{για } i = 1, \dots, n \end{cases}$$

Αυτή η διαδικασία απεικονίζεται στο Σχήμα 3.2, όπου η σύνοψη του προηγούμενου μπλοκ, μαζί με το τρέχον μπλοκ, αποτελούν την είσοδο της συνάρτησης συμπίεσης F . Η σύνοψη (hash) ολόκληρου του μηνύματος



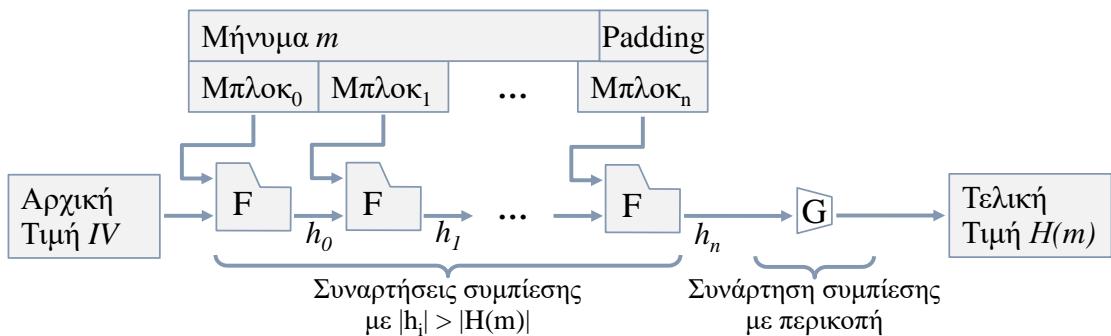
Σχήμα 3.2: Κατασκευή Merkle-Damgård.

είναι το αποτέλεσμα της τελευταίας εφαρμογής της συνάρτησης συμπίεσης. Προαιρετικά, εφαρμόζεται στο τέλος μια συνάρτηση οριστικοποίησης στην τελική τιμή εξόδου $H(m)$ της κατασκευής Merkle-Damgård.

3.2.2 Κατασκευή Wide-Pipe

Η κατασκευή Wide-Pipe [8] είναι παρόμοια με τη κατασκευή Merkle-Damgård (βλέπε Σχήμα 3.3), εκτός από την τιμή σύνοψης h_i που είναι μεγαλύτερη σε μέγεθος από την τελική τιμή σύνοψης $H(m)$ της κατασκευής. Για τον σκοπό αυτό, εφαρμόζεται μια δεύτερη συνάρτηση συμπίεσης για την παραγωγή της τελικής σύνοψης έχοντας ως είσοδο την τελευταία τιμή h_n της αλυσίδας των συναρτήσεων συμπίεσης F . Αυτή η συνάρτηση είναι τόσο απλή όσο μια περικοπή (truncation) ενός υποσυνόλου των bits.

Η κατασκευή Wide-Pipe μετριάζει τις επιθέσεις που βασίζονται σε εσωτερικές συγκρούσεις της ίδιας της κατασκευής, παρέχοντας καλύτερη ασφάλεια από την κατασκευή Merkle-Damgård [9]. Ωστόσο, αυτό έχει ως αποτέλεσμα η εσωτερική κατάσταση της κατασκευής να είναι μεγαλύτερη σε μέγεθος, να απαιτείται περισσότερη μνήμη και δυνητικά περισσότερο υπολογιστικό κόστος για να επιτευχθεί το ίδιο επίπεδο ασφάλειας με μια μικρότερη τιμή σύνοψης h_i .



Σχήμα 3.3: Κατασκευή Wide-Pipe.

3.2.3 Κατασκευή HAIFA

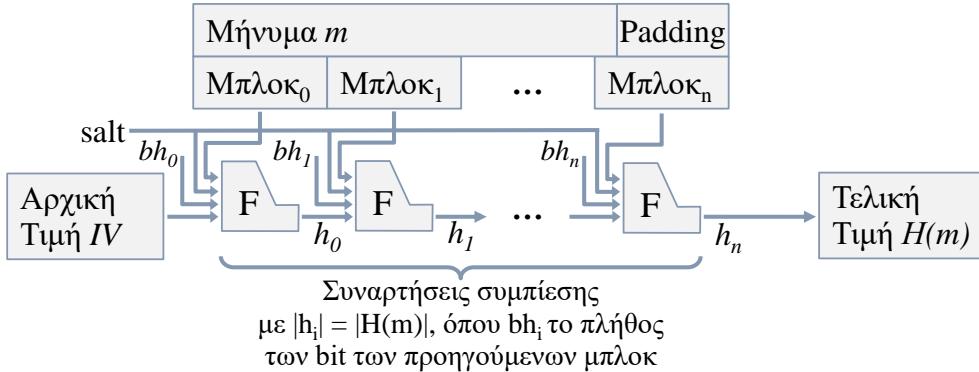
Η κατασκευή HAIFA (HAsh Iterative FrAmework) [10] αποτελεί μια βελτιωμένη έκδοση της κατασκευής Merkle-Damgård (βλέπε Σχήμα 3.4). Οι κύριες διαφορές της από την κατασκευή Merkle-Damgård είναι οι εξής:

- Η συνάρτηση συμπίεσης στην κατασκευή HAIFA λαμβάνει ως πρόσθετες εισόδους μια σταθερή τιμή «αλατιού» (*salt*) και έναν μετρητή bh_i του πλήθους των bit των προηγούμενων μπλοκ.

- Η αρχική τιμή IV και η πλήρωση (padding) εξαρτώνται από το μήκος της σύνοψης, το οποίο είναι μεταβλητό μεγέθους (αλλά όχι μεγαλύτερο από την τιμή σύνοψης h_i).

Η χρήση του μετρητή αποτρέπει διάφορουν τύπου επιθέσεις, όπως η επίθεση επέκτασης μήκους (length extension attack), αντιμετωπίζοντας με αυτό το τρόπο διάφορες αδυναμίες που διαθέτει η κατασκευή Merkle-Damgård [9]. Επιπρόσθετα, η ενσωμάτωση του $salt$ στην συνάρτηση συμπίεσης ενθαρρύνει και απλοποιεί τη χρήση αυτής της κατασκευής για την αποθήκευση κωδικών πρόσβασης και τη δημιουργία ψηφιακών υπογραφών.

Η συνάρτηση σύνοψης BLAKE [9] χρησιμοποιεί μια απλοποιημένη έκδοση της κατασκευής HAIFA, διατηρώντας ωστόσο όλες τις επιθυμητές ιδιότητες της κατασκευής HAIFA.



Σχήμα 3.4: Κατασκευή HAIFA.

3.2.4 Κατασκευή Sponge

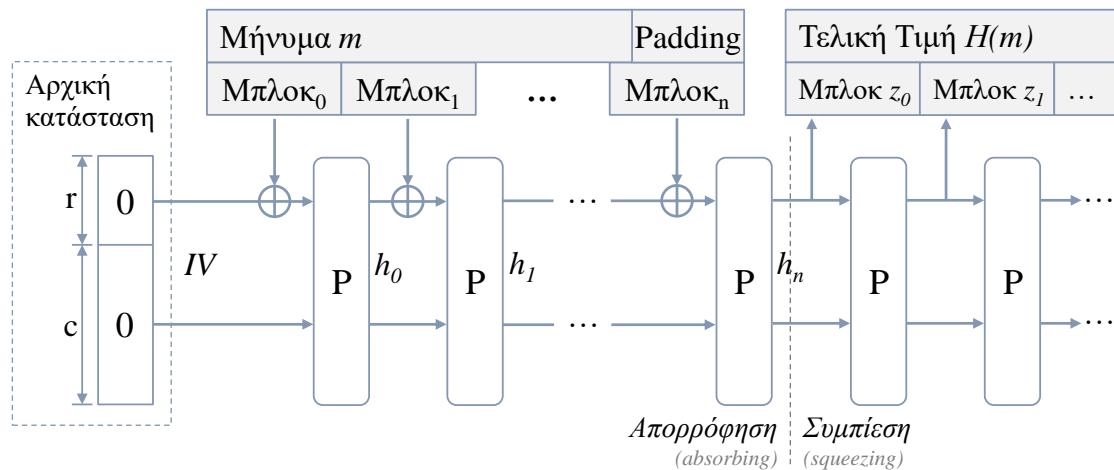
Η κατασκευή (ή συνάρτηση) Sponge [11] αποκλίνει από την κατασκευή Merkle-Damgård όσον αφορά την εφαρμογή των συναρτήσεων συμπίεσης. Πιο συγκεκριμένα, για κάθε μπλοκ δεδομένων του μηνύματος m_i , υπολογίζεται το $P(h_{i-1} \oplus m_i)$, όπου το μπλοκ δεδομένων m_i είναι σημαντικά μικρότερο από την τρέχουσα τιμή σύνοψης h_{i-1} και όπου P είναι μια συνάρτηση ψευδοτυχαίων αντιμεταθέσεων (permutation), η οποία μπορεί να είναι αποτελεσματικά αναστρέψιμη (δηλ., μπορούν να αναφεύθουν σχετικά εύκολα οι αντιμεταθέσεις που πραγματοποιήθηκαν).

Όπως φαίνεται στο Σχήμα 3.5, στην κατασκευή Sponge χρησιμοποιείται μια τιμή εσωτερικής κατάστασης μεγέθους $b = r + c$ bit, που αρχικοποιείται με μηδενικά, όπου:

- Το r ονομάζεται ρυθμός (rate) και αντιστοιχεί στο μέγεθος του κάθε μπλοκ μηνύματος.
- Το c ονομάζεται χωρητικότητα (capacity) και ορίζει το επίπεδο ασφάλειας.

Στη συνέχεια, η κατασκευή Sponge τροποποιεί την εσωτερική της κατάσταση με την εισαγωγή μπλοκ δεδομένων σε συνδυασμό με μια συνάρτηση αντιμεταθέσεων P (permutations). Η κατασκευή Sponge, όπως υπονοεί και το όνομα της, λειτουργεί ως ένα «σφουγγάρι» που σε πρώτη φάση απορροφά (absorbing) τα μπλοκ δεδομένων m_i και σε δεύτερη φάση, μέσω της συμπίεσης (squeezing), αποδεσμεύει σε μορφή μπλοκ z_j τα δεδομένα της τελικής σύνοψης $H(m)$, το πλήθος των οποίων καθορίζεται από την εκάστοτε συνάρτηση σύνοψης.

Ένα από τα πλεονεκτήματα της κατασκευής Sponge είναι η ευελιξία την οποία παρέχει, καθώς διαφοροποιώντας κατάλληλα τις παραμέτρους r και c μπορούν να επιτύχει καλύτερη απόδοση σε σχέση με την ασφάλεια, αλλά και το ανάποδο. Ένα από τα πιο χαρακτηριστικά παραδείγματα χρήσης της κατασκευής Sponge αποτελεί η συνάρτηση σύνοψης SHA-3 [12].



Σχήμα 3.5: Κατασκευή Sponge.

3.2.5 Συναρτήσεις Συμπίεσης με Χρήση Κρυπτογράφησης Μπλοκ

Μια ειδική κατηγορία συναρτήσεων συμπίεσης αποτελούν αυτές των οποίων η λειτουργία τους βασίζεται σε αλγορίθμους κρυπτογράφησης μπλοκ (Ενότητα 1.3). Η χρήσης της κρυπτογράφησης μπλοκ μπορεί να εξασφαλίσει έναν μη αναστρέψιμο μετασχηματισμό των δεδομένων εισόδου, μια ιδιότητα που είναι εξαιρετικά σημαντική στις συναρτήσεις σύνοψης. Οι κύριοι λόγοι χρήσης αυτών των συναρτήσεων συμπίεσης είναι οι εξής:

- **Μεγαλύτερη εμπιστοσύνη:** Η ασφάλεια μιας συνάρτησης σύνοψης μπορεί να διασφαλιστεί με την χρήση μιας καλά καθιερωμένης κρυπτογράφησης μπλοκ παρέχοντας μεγαλύτερη εμπιστοσύνη σε σύγκριση με έναν νέο αλγόριθμο.
- **Μικρότερο μέγεθος υλοποίησης:** Ο κώδικας που χρησιμοποιείται για την κρυπτογράφηση μπλοκ μπορεί να επαναχρησιμοποιηθεί από την συνάρτηση σύνοψης, μειώνοντας έτσι το χώρο που απαιτείται από τα κρυπτογραφικά στοιχεία ενός προγράμματος.
- **Ταχύτητα εκτέλεσης:** Χρησιμοποιώντας, για παράδειγμα, τον αλγόριθμο κρυπτογράφησης μπλοκ AES (Ενότητα 1.3.2), μπορούν να επιτευχθούν αρκετά μεγάλες ταχύτητες εκτέλεσης των συναρτήσεων σύνοψης, γιατί η υλοποίηση του AES παρέχεται σε επίπεδο υλικού από σχεδόν όλους τους σύγχρονους επεξεργαστές.

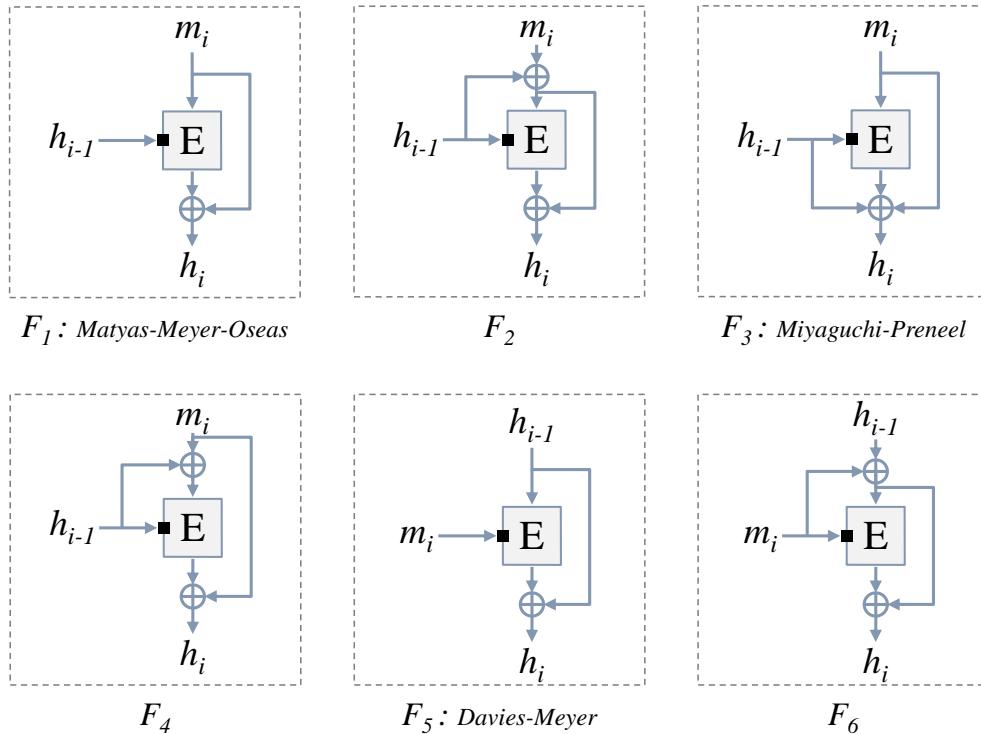
Ωστόσο υπάρχουν και κάποια μειονεκτήματα σχετικά με τη χρήση κρυπτογράφησης μπλοκ σε συναρτήσεις σύνοψης:

- **Διαρθρωτικά προβλήματα:** Γενικά το μέγεθος του μπλοκ και του κλειδιού των μπλοκ κρυπτοσυστημάτων δεν είναι σε συμφωνία με τις τιμές που απαιτούνται στις συναρτήσεις σύνοψης. Για παράδειγμα, ο AES χρησιμοποιεί μπλοκ μεγέθους 128 bits, ενώ μια συνάρτηση σύνοψης μπορεί να επιστρέψει σύνοψεις τουλάχιστον 224 bits. Συνεπώς, θα πρέπει να χρησιμοποιηθούν κατασκευές με πολλαπλά στιγμιότυπα (instances) κρυπτογράφησης μπλοκ, καθιστώντας έτσι τις συναρτήσεις σύνοψης λιγότερο αποδοτικές.
- **Αργό χρονοδιάγραμμα δημιουργίας κλειδιών:** Η αρχικοποίηση της κρυπτογράφησης μπλοκ με την δημιουργία κατάλληλης χρονοδρομολόγησης κλειδιών (key schedule) είναι συνήθως αργή, γεγονός που έχει ως αποτέλεσμα τη χρήση καθορισμένων μεταθέσεων κλειδιού και όχι μιας πιο σύνθετης μορφής μεταθέσεων. Ωστόσο, κάτι τέτοιο έχει ως αποτέλεσμα οι συναρτήσεις συμπίεσης να μην είναι αρκετά αποδοτικές και ασφαλείς [13].

Στο Σχήμα 3.6 απεικονίζονται διάφορες γνωστές συναρτήσεις συμπίεσης που βασίζονται σε κρυπτοσυστήματα μπλοκ [14]. Στο σχήμα αυτό θα πρέπει να σημειωθεί ότι η συνάρτηση κρυπτογράφησης E έχει ως κλειδί την τιμή που εισάγεται στο σημείο ■, ενώ επιπλέον έχει γίνει η παραδοχή ότι τα κλειδιά και τα μπλοκ δεδομένων είναι του ίδιου μεγέθους. Οι έξι συναρτήσεις συμπίεσης που απεικονίζονται σε αυτό το σχήμα δίνονται αντίστοιχα από τις ακόλουθες εξισώσεις:

$$\begin{aligned} F_1 &: E_{h_{i-1}}(m_i) \oplus m_i \\ F_2 &: E_{h_{i-1}}(h_{i-1} \oplus m_i) \oplus h_{i-1} \oplus m_i \\ F_3 &: E_{h_{i-1}}(m_i) \oplus h_{i-1} \oplus m_i \\ F_4 &: E_{h_{i-1}}(h_{i-1} \oplus m_i) \oplus m_i \\ F_5 &: E_{m_i}(h_{i-1}) \oplus h_{i-1} \\ F_6 &: E_{m_i}(h_{i-1} \oplus m_i) \oplus h_{i-1} \oplus m_i \end{aligned}$$

όπου το $E_x(y)$ υποδηλώνει την κρυπτογράφηση του μπλοκ δεδομένων y με το x ως κλειδί κρυπτογράφησης. Από αυτές τις συναρτήσεις συμπίεσης, η συνάρτηση F_1 είναι γνωστή ως κατασκευή Matyas-Meyer-Oseas, η συνάρτηση F_3 είναι γνωστή ως κατασκευή Miyaguchi-Preneel και χρησιμοποιείται στην συνάρτηση σύνοψης Whirlpool, και τέλος η συνάρτηση F_5 είναι γνωστή ως κατασκευή Davies-Meyer και χρησιμοποιείται στις συναρτήσεις σύνοψης MD5, SHA-1, και SHA-2.



Σχήμα 3.6: Συναρτήσεις συμπίεσης που βασίζονται σε κρυπτογράφηση μπλοκ.

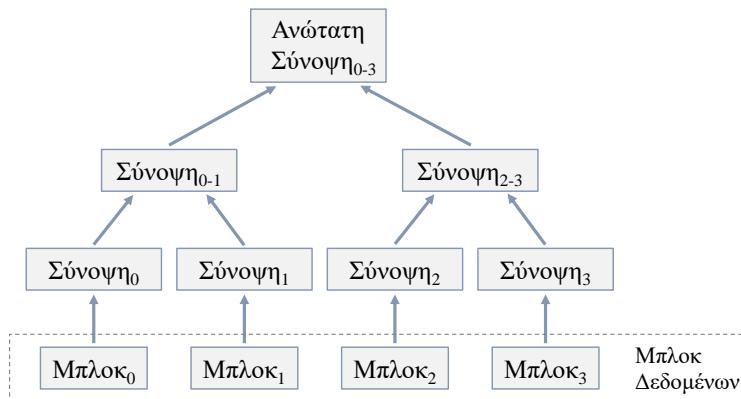
3.2.6 Δένδρα Merkle

Ένα δέντρο Merkle [15] ή απλά ένα δέντρο σύνοψης, είναι ένα δέντρο στο οποίο κάθε κόμβος «φύλλο» είναι το αποτέλεσμα σύνοψης ενός μπλοκ δεδομένων και κάθε άλλος κόμβος σε αυτό είναι το αποτέλεσμα σύνοψης των θυγατρικών του κόμβων (δηλ. των επιμέρους συνόψεων). Τα δέντρα Merkle επιτρέπουν κατά κύριο λόγο την αποτελεσματική και ασφαλή επαλήθευση του περιεχομένου μεγάλων δομών δεδομένων (π.χ.

ένα μεγάλο αρχείο ή ένα σύνολο αρχείων). Για παράδειγμα, στο Σχήμα 3.7, η Σύνοψη₀₋₁ είναι το αποτέλεσμα μιας συνάρτησης σύνοψης από την συνένωση της Σύνοψη₀ και της Σύνοψη₁. Δηλαδή, $\Sigma\text{ύνοψη}_{0-1} = H(\Sigma\text{ύνοψη}_0 || \Sigma\text{ύνοψη}_1)$, όπου H είναι μια συνάρτηση σύνοψης και $||$ δηλώνει συνένωση.

Οι περισσότερες υλοποιήσεις των δένδρων Merkle είναι δυαδικές (δύο κόμβοι-παιδιά βρίσκονται κάτω από κάθε κόμβο-απόγονο), αλλά μπορούν επίσης να χρησιμοποιούν περισσότερους κόμβους-παιδιά κάτω από κάθε κόμβο. Συνήθως, για τον υπολογισμό των συνόψεων ενός δέντρου Merkle χρησιμοποιείται μια συνάρτηση σύνοψης, όπως η SHA-2.

Στην κορυφή ενός δέντρου Merkle υπάρχει η ανώτατη σύνοψη (top hash) (ή αλλιώς root ή master hash). Για παράδειγμα, πριν από τη λήψη ενός μεγάλου αρχείου από ένα δίκτυο ομότιμων κόμβων (peer-to-peer network), στις περισσότερες περιπτώσεις η ανώτατη σύνοψη αποκτάται από μια αξιόπιστη πηγή (π.χ. έναν φίλο ή μια έμπιστη τοποθεσία διαδικτύου) ή ακόμη και ολόκληρο το δέντρο Merkle ενσωματώνοντας όλες τις ενδιάμεσες συνόψεις. Όταν είναι διαθέσιμη η ανώτατη σύνοψη, τα μπλοκ δεδομένων του αρχείου μπορούν να ληφθούν από οποιαδήποτε μη αξιόπιστη πηγή, όπως έναν οποιονδήποτε κόμβο του δικτύου ομότιμων κόμβων. Στη συνέχεια, παράγεται το δέντρο Merkle βάσει των μπλοκ δεδομένων που λήφθηκαν, και ελέγχεται έναντι της έμπιστης ανώτατης σύνοψης. Εάν το δέντρο που θα προκύψει είναι πλαστό, θα δοκιμαστεί να ληφθούν μπλοκ δεδομένων από μια άλλη πηγή έως ότου να παραχθεί ένα δέντρο Merkle που να αντιστοιχεί στην ανώτατη σύνοψη.



Σχήμα 3.7: Παράδειγμα ενός δυαδικού δέντρου Merkle με τέσσερα μπλοκ δεδομένων.

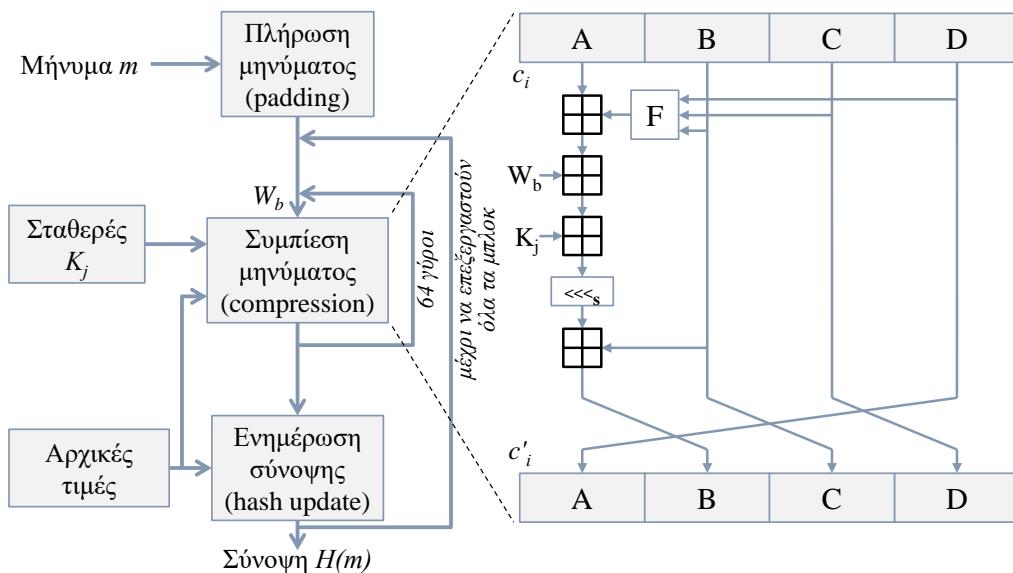
Η κύρια διαφορά ενός δέντρου Merkle από μια λίστα συνόψεων (hash list) ή αλυσίδα συνόψεων (hash chain) [16], κατά την οποία όλες οι επιμέρους συνόψεις υπολογίζονται σειριακά, είναι ότι ένας κλάδος του δέντρου μπορεί να μεταφορτωθεί ανά πάσα στιγμή και η ακεραιότητα του κάθε κλάδου μπορεί να ελεγχθεί άμεσα, ακόμα κι αν ολόκληρο το δέντρο δεν είναι ακόμη διαθέσιμο. Για παράδειγμα, στο Σχήμα 3.7, η ακεραιότητα του $Mπλοκ_1$ μπορεί να επαληθευτεί άμεσα εάν το δέντρο περιέχει ήδη την $\Sigma\text{ύνοψη}_0$ και την $\Sigma\text{ύνοψη}_{2-3}$. Για να γίνει αυτό, υπολογίζουμε την σύνοψη του $Mπλοκ_1$, την συνδυάζουμε με την $\Sigma\text{ύνοψη}_0$ και κατόπιν με την $\Sigma\text{ύνοψη}_{2-3}$, και στο τέλος συγκρίνουμε το αποτέλεσμα υπολογισμού με την Ανώτατη $\Sigma\text{ύνοψη}_{0-3}$. Παρόμοια, η ακεραιότητα του $Mπλοκ_2$ μπορεί να επαληθευτεί εάν το δέντρο περιέχει ήδη την $\Sigma\text{ύνοψη}_3$ και την $\Sigma\text{ύνοψη}_{0-1}$. Όπως γίνεται αντιληπτό, μια τέτοια διαδικασία παρουσιάζει πλεονεκτήματα, αφού είναι αποτελεσματικότερο να τεμαχίσουμε ένα αρχείο σε μικρά μπλοκ δεδομένων, και εάν υποστεί βλάβη κάποιο από αυτά, τότε χρειάζεται να μεταφορτωθεί μόνο αυτό το μπλοκ που καταστράφηκε.

Τέλος, μια σχετικά σύγχρονη εφαρμογή των δένδρων Merkle είναι η χρήση του ως κατασκευή στην συνάρτηση σύνοψης BLAKE3 [17]. Η συνάρτηση σύνοψης BLAKE3 είναι εξαιρετικά γρήγορη σε σύγκριση με άλλες συναρτήσεις σύνοψης [17] και αυτό το οφείλει στο γεγονός ότι η συνένωση των συνόψεων των μπλοκ δεδομένων δεν γίνεται σειριακά (όπως π.χ. στις κατασκευές που παρουσιάστηκαν στις Ενότητες 3.2.1 - 3.2.4), αλλά παράλληλα αξιοποιώντας την δομή των δένδρων Merkle.

3.3 Συνάρτηση Σύνοψης MD5

Η συνάρτηση σύνοψης MD5 (Message-Digest algorithm 5) [18], προτάθηκε για πρώτη φορά από τον Ronald Rivest το 1991 (για να αντικαταστήσει τον MD4) και χρησιμοποιείται μέχρι και σήμερα, κυρίως για τον έλεγχο της ακεραιότητας αρχείων που διακινούνται μέσω διαδικτύου. Ωστόσο, η συνάρτηση MD5 θεωρείται ότι δεν είναι τόσο ανθεκτική σε συγκρούσεις (collision resistant) [19] και δεν ενδείκνυται για σοβαρές κρυπτογραφικές εφαρμογές, όπως οι ψηφιακές υπογραφές.

Η λειτουργία της συνάρτησης σύνοψης MD5 βασίζεται στην κατασκευή Merkle-Damgård (Ενότητα 3.2.1) και σε υψηλό επίπεδο μπορεί να χωριστεί σε τρεις κύριες περιοχές: την πλήρωση (padding) μηνύματος, την συμπίεση (compression) μηνύματος και την ενημέρωση (update) σύνοψης. Στο Σχήμα 3.8 αποτυπώνεται η περιγραφή της λειτουργίας της συνάρτησης σύνοψης MD5 και ακολουθεί η αναλυτική περιγραφή της.



Σχήμα 3.8: Λειτουργία της συνάρτησης σύνοψης MD5.

Αρχικά, το μήνυμα χωρίζεται σε μπλοκ δεδομένων μήκους 512 bits και πραγματοποιείται πλήρωση (padding) του τελευταίου μπλοκ του μηνύματος, έτσι ώστε το μέγεθος του να γίνει 512 bits. Η διαδικασία πλήρωσης γίνεται ως εξής:

- Αρχικά, προστίθεται στο τέλος των δεδομένων ένα bit με τιμή “1”, για να σηματοδοτήσει την αρχή του επιθέματος (postfix).
- Στη συνέχεια, προστίθενται τόσα bits με τιμή “0”, όσα είναι απαραίτητα έτσι ώστε το τρέχον μήκος των δεδομένων του μπλοκ να είναι ίσο με 448 bits.
- Τέλος, προσαρτάται η τιμή (με μέγεθος 64-bits) του συνολικού μήκους του μηνύματος πριν την πλήρωσή του.

Κατά την συμπίεση (compression) του μηνύματος, ο συνάρτηση σύνοψης MD5 λειτουργεί σε μια κατάσταση συμπίεσης c_i μεγέθους 128 bits που χωρίζεται σε τέσσερις λέξεις, A , B , C και D , των 32 bits η κάθε μια. Αρχικά, η κατάσταση αυτή αρχικοποιείται με προκαθορισμένες αρχικές τιμές που ορίζει η συνάρτηση σύνοψης MD5. Η συμπίεση μηνύματος πραγματοποιείται κάθε φορά για ένα μπλοκ δεδομένων 512 bits και με βάση αυτό το μπλοκ τροποποιείται η τρέχουσα κατάσταση σε 4 γύρους, με κάθε γύρο να αποτελείται από μια επαναληπτική διαδικασία για τις 16 λέξεις (32 bits) του μπλοκ. Συνολικά, πραγματοποιούνται 64 γύροι (4 x 16) και η λειτουργία του κάθε γύρου απεικονίζεται σε μεγέθυνση στο δεξί μέρος του Σχήματος 3.8. Στο σχήμα

αυτό, το W_b υποδηλώνει μια λέξη 32 bits του μπλοκ, το K_j υποδηλώνει μια σταθερά 32 bits διαφορετική για κάθε γύρο, το σύμβολο \lll_s υποδηλώνει μια ολίσθηση προς τα αριστερά κατά s θέσεις bit (το s διαφέρει σε κάθε γύρο), το σύμβολο \oplus υποδηλώνει μια πρόσθεση με μόντουλο 2^{32} , ενώ το F υποδηλώνει μια διαφορετική μη γραμμική συνάρτηση για κάθε έναν από τους 4 γύρους:

$$\begin{aligned} F_1(B, C, D) &= (B \wedge C) \vee ((\neg B) \wedge D) \\ F_2(B, C, D) &= (B \wedge D) \vee (C \wedge (\neg D)) \\ F_3(B, C, D) &= B \oplus C \oplus D \\ F_4(B, C, D) &= C \oplus (B \vee (\neg D)) \end{aligned}$$

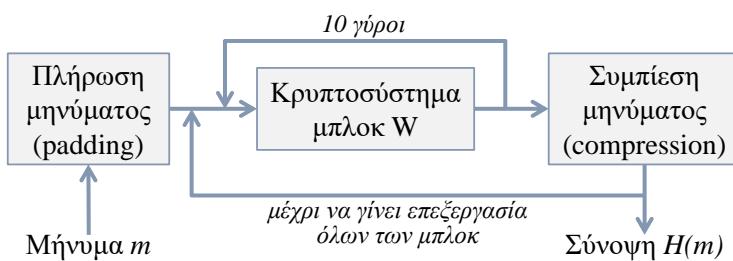
όπου \oplus , \wedge , \vee και \neg υποδηλώνουν τις πράξεις XOR, AND, OR και NOT, αντίστοιχα.

Τέλος, κατά την ενημέρωση (update) σύνοψης, η συνάρτηση σύνοψης MD5 ορίζει την αρχική κατάσταση σύνοψης h_0 μεγέθους 128 bits με βάση προκαθορισμένες αρχικές τιμές (τις ίδιες με την συμπίεση μηνύματος), και μετά το τέλος της συμπίεσης ενός μπλοκ, προσθέτει στην τρέχουσα κατάσταση σύνοψης h_i την κατάσταση συμπίεσης c'_i που προέκυψε από το τρέχον μπλοκ δεδομένων m_i , δηλ. $h_{i+1} = h_i + c'_i$. Η τελική σύνοψη $H(m)$ του αλγορίθμου MD5 αποτελείται από την συνένωση των τεσσάρων λέξεων της τελικής κατάστασης σύνοψης h_n , δηλ. $H(m) = h_n^A \parallel h_n^B \parallel h_n^C \parallel h_n^D$, όπου n το πλήθος των μπλοκ.

3.4 Συνάρτηση Σύνοψης Whirlpool

Η συνάρτηση σύνοψης Whirlpool [20] αναπτύχθηκε το 2000 από τους Vincent Rijmen και Paulo Barreto. Η συνάρτηση σύνοψης Whirlpool ήταν μία από τις δύο που επιλέχθηκαν (η άλλη ήταν η SHA-2), με βάση την ασφάλεια που παρέχουν, στα πλαίσια του έργου NESSIE (New European Schemes for Signatures, Integrity, and Encryption) [21]. Πιο συγκεκριμένα, στα πλαίσια του έργου NESSIE ζητήθηκε να αναπτυχθούν νέες συναρτήσεις σύνοψης και αυτές να υποβληθούν σε αυστηρή τριετή αξιολόγηση και κρυπτογραφική ανάλυση. Επίσης, η συνάρτηση σύνοψης Whirlpool έχει συμπεριληφθεί στο πρότυπο ISO/IEC 10118-3 [22].

Η λειτουργία της συνάρτησης σύνοψης Whirlpool συνοψίζεται στο Σχήμα 3.9. Όπως φαίνεται σε αυτό το σχήμα, η λειτουργία της Whirlpool μπορεί να χωριστεί σε τρεις κύριες περιοχές: την πλήρωση (padding) μηνύματος, το κρυπτοσύστημα μπλοκ W και την συμπίεση (compression) μηνύματος.



Σχήμα 3.9: Λειτουργία της συνάρτησης σύνοψης Whirlpool.

Η πλήρωση (padding) μηνύματος στην συνάρτηση σύνοψης Whirlpool πραγματοποιείται με τα ακόλουθα τρία βήματα:

- Αρχικά, προστίθεται στο τέλος του μηνύματος ένα bit με τιμή “1”.
- Στη συνέχεια, προστίθενται τόσα bits με τιμή “0”, όσα είναι απαραίτητα έτσι ώστε το μήκος του μηνύματος να είναι πολλαπλάσιο του 256.

- Τέλος, προσαρτάται η τιμή του συνολικού μήκους του μηνύματος πριν την πλήρωσή του, ως ένας unsigned ακέραιος αριθμός 256 bits.

Μόλις ολοκληρωθεί η πλήρωση του μηνύματος, κάθε μπλοκ δεδομένων 512 bits οργανώνεται ως ένας πίνακας 8×8 bytes. Για την επεξεργασία ενός μόνο μπλοκ δεδομένων, εφαρμόζονται 10 επαναλήψεις του κρυπτοσυστήματος μπλοκ W [20]. Το κρυπτοσύστημα W βασίζεται στη δομή του κρυπτοσυστήματος AES (Ενότητα 1.3.2) και εσωτερικά αποτελείται από τέσσερις μετασχηματισμούς, το SubBytes, το ShiftColumns, το MixRows και το AddRoundKey, με παρόμοιες λειτουργίες όπως και στον AES.

Μετά από τους 10 γύρους εκτέλεσης του κρυπτοσυστήματος W , εκτελείται το σχήμα συμπίεσης Miyaguchi-Preneel (Σχήμα 3.6, Ενότητα 3.2.5). Σε ένα τέτοιο σχήμα απαιτούνται δύο είσοδοι, το m_i και το h_{i-1} , όπου το m_i είναι το μπλοκ δεδομένων προς συμπίεση και το h_{i-1} είναι η έξοδος της προηγούμενης επανάληψης του σχήματος συμπίεσης Miyaguchi-Preneel. Οπότε, όπως φαίνεται στο Σχήμα 3.6(F_3), η έξοδος h_i της τρέχουσας επανάληψης, εξαρτάται από το μπλοκ δεδομένων m_i , την έξοδο της προηγούμενης επανάληψης h_{i-1} , και την έξοδο $E_{h_{i-1}}$ του κρυπτοσυστήματος μπλοκ W . Πιο συγκεκριμένα, η πράξη η οποία εκτελείται δίνεται από την σχέση: $h_i = E_{h_{i-1}}(m_i) \oplus h_{i-1} \oplus m_i$. Αυτή η διαδικασία επαναλαμβάνεται μέχρι να επεξεργαστούν όλα τα μπλοκ του μηνύματος.

3.5 Οικογένεια Συναρτήσεων Σύνοψης SHA

Η οικογένεια συναρτήσεων σύνοψης SHA (Secure Hash Algorithm) αποτελεί ένα σύνολο συναρτήσεων σύνοψης που έχουν δημοσιευθεί από το National Institute of Standards and Technology (NIST) ως πρότυπα (FIPS) των Ηνωμένων Πολιτειών. Σε αυτήν την οικογένεια συναρτήσεων σύνοψης συμπεριλαμβάνονται μέχρι στιγμής τέσσερα μέλη, οι συναρτήσεις SHA-0, SHA-1, SHA-2 και SHA-3. Στον Πίνακα 3.1 γίνεται μια (με αναφορά στη συνάρτηση MD5) παρουσίαση των τεσσάρων αυτών μελών της οικογένειας SHA.

Πίνακας 3.1: Συγκριτική παρουσίαση της οικογένειας συναρτήσεων σύνοψης SHA.

Όνομα Αλγορίθμου	Μέγεθος Εξόδου (bits)	Μέγεθος Εσωτερικής Κατάστασης (bits)	Μέγεθος Μπλοκ Εισόδου (bits)	Γύροι	Πράξεις*	Ημερομηνία Δημοσίευσης
MD5 (ως αναφορά)	128	128 (4×32)	512	64	And, Xor, Rot, Or, Add (mod 2^{32})	1992
SHA-0	160	160 (5×32)	512	80	And, Xor, Rot, Or, Add (mod 2^{32})	1993
SHA-1	160	160 (5×32)	512	80	And, Xor, Rot, Or, Add (mod 2^{32})	1995
SHA-2	SHA-224 SHA-256 SHA-384 SHA-512 SHA-512/224 SHA-512/256	224 256 384 512 224 256	256 (8×32) 512 (8×64)	512 1024	64 80 And, Xor, Rot, Shr, Or, Add (mod 2^{32}) And, Xor, Rot, Shr, Or, Add (mod 2^{64})	2004 2001 - - - - 2012 - -
SHA-3	SHA3-224 SHA3-256 SHA3-384 SHA3-512 SHAKE128 SHAKE256	224 256 384 512 $d_{(\text{αυθαίρετο})}$ $d_{(\text{αυθαίρετο})}$	1600 ($5 \times 5 \times 64$)	1152 1088 832 576 1344 1088	24	And, Xor, Rot, Not - - - - - - - - - -

*όπου Rot και Shr υποδηλώνουν ολισθηση (rotate no carry) και αριστερή λογική μετατόπιση (right logical shift), αντίστοιχα.

Το πρώτο μέλος της οικογένειας, δημοσιεύθηκε το 1993 και επίσημα ονομάζεται SHA [23]. Ωστόσο, κα-

λείται συχνά ως SHA-0 για να αποφευχθεί η σύγχυση με τους διαδόχους του. Η συνάρτηση σύνοψης SHA-0 αποσύρθηκε λίγο μετά τη δημοσίευση του λόγω ενός αδιευκρίνιστου προβλήματος που παρουσιάστηκε και αντικαταστάθηκε από την ελαφρώς αναθεωρημένη έκδοση SHA-1 [24].

3.5.1 Συνάρτηση Σύνοψης SHA-1

Η συνάρτηση σύνοψης SHA-1 [24] δημοσιεύθηκε το 1995 από την Υπηρεσία Εθνικής Ασφάλειας (NSA) των Ηνωμένων Πολιτειών. Βασίζεται σε αρχές παρόμοιες με αυτές που χρησιμοποιήθηκαν κατά τη σχεδίαση του MD5 (Ενότητα 3.3) και ουσιαστικά θεωρείται ο διάδοχός του. Η συνάρτηση SHA-1 έχει ως έξοδο σύνοψη μεγέθους 160 bits. Το μεγαλύτερο μέγεθος της παραγόμενης σύνοψης κάνει την SHA-1 ασφαλέστερη από την συνάρτηση MD5, αν και μια σημαντική διαφορά μεταξύ των δυο συναρτήσεων αφορά το μέγεθος της εσωτερικής τους κατάστασης. Η συνάρτηση SHA-1 αποτέλεσε μια αρκετά δημοφιλή συνάρτηση σύνοψης και υιοθετήθηκε σε μια μεγάλη ποικιλία δημοφιλών εφαρμογών και πρωτοκόλλων ασφάλειας, συμπεριλαμβανομένων του TLS, SSL, PGP, SSH, S/MIME, και IPSec.

Η λειτουργία της συνάρτησης σύνοψης SHA-1 βασίζεται στην κατασκευή Merkle-Damgård (Ενότητα 3.2.1) και σε υψηλό επίπεδο μπορεί να χωριστεί σε τέσσερις κύριες περιοχές: την πλήρωση (padding) μηνύματος, την επέκταση (expansion) μηνύματος, την συμπίεση (compression) μηνύματος, και την ενημέρωση (update) σύνοψης. Στο Σχήμα 3.10 αποτυπώνεται η περιγραφή της λειτουργίας της συνάρτησης σύνοψης SHA-1 και ακολουθεί η αναλυτική περιγραφή της.

Αρχικά, το μήνυμα χωρίζεται σε μπλοκ δεδομένων μήκους 512 bits και πραγματοποιείται πλήρωση (padding) του τελευταίου μπλοκ του μηνύματος, έτσι ώστε το μέγεθος αυτού του μπλοκ να γίνει 512 bits. Η διαδικασία πλήρωσης γίνεται ως εξής:

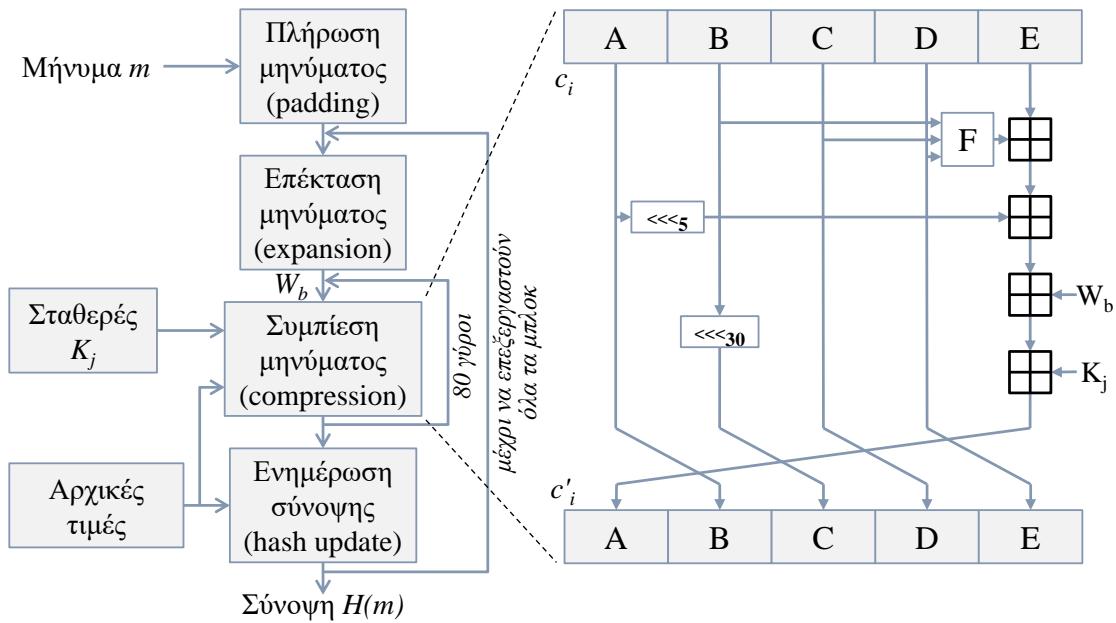
1. Προστίθεται ένα bit με τιμή “1” στο τέλος του μηνύματος.
2. Στη συνέχεια, προστίθενται k bits με τιμή “0”, όπου $k \geq 0$ και αποτελεί την λύση με τη μικρότερη τιμή της εξίσωσης $l + 1 + k \equiv 448 \pmod{512}$.
3. Τέλος, προσαρτάται η τιμή (ως unsigned big-endian μεγέθους 64-bits) του συνολικού μήκους του μηνύματος των l -bits πριν την πλήρωσή του.

Κατά την επέκταση μηνύματος, παράγονται δεδομένα μεγέθους μεγαλύτερου από το μέγεθος του μπλοκ, πραγματοποιώντας μια σειρά πράξεων στα δεδομένα που περιέχονται στο τρέχον μπλοκ m_i μεγέθους 512 bits. Οι πράξεις αυτές αποτυπώνονται στην ακόλουθη εξίσωση:

$$W_b = \begin{cases} m_i^b & 0 \leq b \leq 15 \\ (W_{b-3} \oplus W_{b-8} \oplus W_{b-14} \oplus W_{b-16}) \lll_1 & 16 \leq b \leq 79 \end{cases}$$

όπου b ο αριθμός του γύρου και τα σύμβολα \oplus και \lll_1 υποδηλώνουν την πράξη XOR και την αριστερή ολίσθηση κατά μια θέση bit, αντίστοιχα. Αυτό έχει ως αποτέλεσμα την παραγωγή δεδομένων μεγέθους 2560 bits, αφού κάθε λέξη W_b έχει μήκος 32 bits και συνολικά υπάρχουν 80 γύροι. Επιπλέον, θα πρέπει να σημειωθεί ότι οι πρώτες 16 λέξεις της επέκτασης μηνύματος είναι απλά λέξεις 32 bits του μπλοκ δεδομένων.

Κατά την συμπίεση μηνύματος, η συνάρτηση σύνοψης SHA-1 λειτουργεί σε μια κατάσταση συμπίεσης c_i μεγέθους 160 bits που χωρίζεται σε πέντε λέξεις, A, B, C, D και E , των 32 bits η κάθε μια. Αρχικά, η κατάσταση αυτή αρχικοποιείται με προκαθορισμένες αρχικές τιμές που ορίζει η συνάρτηση σύνοψης SHA-1. Η συμπίεση μηνύματος πραγματοποιείται κάθε φορά για ένα μπλοκ δεδομένων 512 bits και με βάση τα δεδομένα επέκτασης μηνύματος αυτού του μπλοκ τροποποιείται η τρέχουσα κατάσταση σε συνολικά 80 γύρους, όπως απεικονίζεται στο δεξιό μέρος του Σχήματος 3.10. Στο σχήμα αυτό, το W_b υποδηλώνει μια λέξη 32 bits της επέκτασης μηνύματος, το K_j υποδηλώνει μια σταθερά 32 bits, τα \lll_5 και \lll_{30} υποδηλώνουν μια αριστερή ολίσθηση κατά 5 και 30 bit, αντίστοιχα, το \oplus υποδηλώνει μια πρόσθεση με μόντουλο 2^{32} , και το F υποδηλώνει μια μη γραμμική συνάρτηση που δίνεται από την ακόλουθη εξίσωση:



Σχήμα 3.10: Λειτουργία της συνάρτησης σύνοψης SHA-1.

$$F(B, C, D) = \begin{cases} (B \wedge C) \vee ((\neg B) \wedge D) & 0 \leq b \leq 19 \\ B \oplus C \oplus D & 20 \leq b \leq 39 \\ (B \wedge C) \vee (B \wedge D) \vee (C \wedge D) & 40 \leq b \leq 59 \\ B \oplus C \oplus D & 60 \leq b \leq 79 \end{cases}$$

όπου b ο αριθμός του γύρου και τα σύμβολα \oplus , \wedge , \vee και \neg υποδηλώνουν τις πράξεις XOR, AND, OR και NOT, αντίστοιχα.

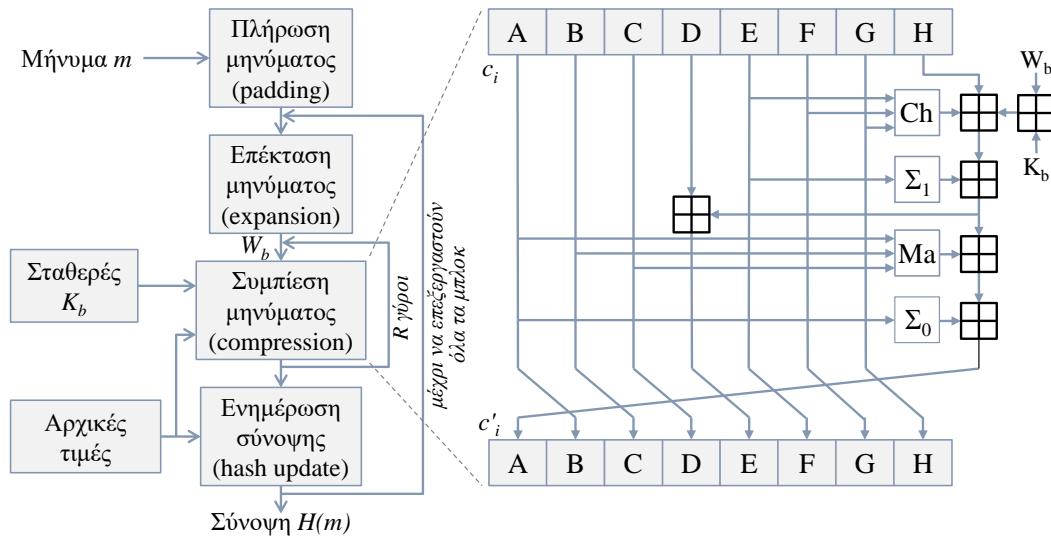
Τέλος, κατά την ενημέρωση σύνοψης, η συνάρτηση σύνοψης SHA-1 ορίζει την αρχική κατάσταση σύνοψης h_0 μεγέθους 160 bits με βάση προκαθορισμένες αρχικές τιμές (τις ίδιες με την συμπίεση μηνύματος), ενώ μετά το τέλος της συμπίεσης ενός μπλοκ, προσθέτει στην τρέχουσα κατάσταση σύνοψης h_i την κατάσταση συμπίεσης c'_i που προέκυψε από το τρέχον μπλοκ m_i , δηλ. $h_{i+1} = h_i + c'_i$. Η τελική σύνοψη $H(m)$ του αλγορίθμου SHA-1 αποτελείται από την συνένωση των πέντε λέξεων της τελικής κατάστασης σύνοψης h_n , δηλ. $H(m) = h_n^A \parallel h_n^B \parallel h_n^C \parallel h_n^D \parallel h_n^E$, όπου n το πλήθος των μπλοκ.

3.5.2 Συνάρτηση Σύνοψης SHA-2

Η συνάρτηση σύνοψης SHA-2 [25] δημοσιεύθηκε για πρώτη φορά το 2001 από την Υπηρεσία Εθνικής Ασφαλείας (NSA) των Ηνωμένων Πολιτειών. Η γενιά συναρτήσεων SHA-2 περιλαμβάνει κατά κύριο λόγο τις συναρτήσεις σύνοψης SHA-224, SHA-256, SHA-384 και SHA-512, ενώ το 2012 προτάθηκαν οι SHA-512/224 και SHA-512/256. Οι SHA-256 και SHA-512 είναι συναρτήσεις σύνοψης που υπολογίζονται με οκτώ λέξεις 32 bits και 64 bits, αντίστοιχα. Χρησιμοποιούν διαφορετικές τιμές ολίσθησης και προσθήκης σταθερών, αλλά η δομή τους είναι κατά τα άλλα σχεδόν ίδια, διαφέροντας μόνο στον αριθμό των γύρων (64 και 80 αντίστοιχα). Οι SHA-224 και SHA-384 είναι τροποποιημένες εκδόσεις των SHA-256 και SHA-512 αντίστοιχα, υπολογισμένες με διαφορετικές αρχικές τιμές. Οι SHA-512/224 και SHA-512/256 είναι επίσης τροποποιημένες εκδόσεις του SHA-512, αλλά οι αρχικές τιμές δημιουργούνται χρησιμοποιώντας τη μέθοδο που περιγράφεται στο FIPS PUB 180-4 [7].

Όπως φαίνεται στο Σχήμα 3.11, η λειτουργία των μελών της οικογένειας συναρτήσεων SHA-2 είναι αρκετά όμοια με αυτή του SHA-1, ακολουθώντας ακριβώς τα ίδια βήματα για την πλήρωση (padding) μηνύ-

ματος, και έχοντας μόνο διαφορά στις συναρτήσεις σύνοψης SHA-384, SHA-512, SHA-512/224 και SHA-512/256, όπου το μέγεθος του μπλοκ είναι 1024 bits και το k θα πρέπει να ικανοποιεί την εξίσωση $l+1+k \equiv 896 \pmod{1024}$, με το l να αναπαριστάται ως unsigned big-endian μεγέθους 128-bits.



Σχήμα 3.11: Λειτουργία της συνάρτησης σύνοψης SHA-2.

Κατά την επέκταση μηνύματος, παράγονται δεδομένα, μεγέθους μεγαλύτερου από το μέγεθος του μπλοκ (512 ή 1024 bits), πραγματοποιώντας μια σειρά πράξεων στα δεδομένα που περιέχονται στο τρέχον μπλοκ m_i . Οι πράξεις αυτές αποτυπώνονται στην ακόλουθη εξίσωση:

$$W_b = \begin{cases} m_i^b & 0 \leq b \leq 15 \\ W_{b-16} + s_0 + W_{b-7} + s_1 & 16 \leq b \leq 63 \text{ ή } 79 \end{cases}$$

και με s_0 και s_1 να δίνονται από τις ακόλουθες εξισώσεις:

$$s_0 = \begin{cases} (W_{b-15} \ggg_7) \oplus (W_{b-15} \ggg_{18}) \oplus (W_{b-15} \ggg_3) & \text{μπλοκ = 512 bits, } 16 \leq b \leq 63 \\ (W_{b-15} \ggg_1) \oplus (W_{b-15} \ggg_8) \oplus (W_{b-15} \ggg_7) & \text{μπλοκ = 1024 bits, } 16 \leq b \leq 79 \end{cases}$$

$$s_1 = \begin{cases} (W_{b-2} \ggg_{17}) \oplus (W_{b-2} \ggg_{19}) \oplus (W_{b-2} \ggg_{10}) & \text{μπλοκ = 512 bits, } 16 \leq b \leq 63 \\ (W_{b-2} \ggg_{19}) \oplus (W_{b-2} \ggg_{61}) \oplus (W_{b-2} \ggg_6) & \text{μπλοκ = 1024 bits, } 16 \leq b \leq 79 \end{cases}$$

όπου b ο αριθμός του γύρου και τα σύμβολα \oplus και \ggg_x υποδηλώνουν την πράξη XOR και μια δεξιά ολίσθηση κατά x bit, αντίστοιχα. Επιπλέον, θα πρέπει να σημειωθεί ότι οι πρώτες 16 λέξεις της επέκτασης μηνύματος είναι απλά λέξεις μεγέθους 32 ή 64 bit, ανάλογα με το μέγεθος του μπλοκ.

Κατά την συμπίεση μηνύματος, η συνάρτηση σύνοψης SHA-1 λειτουργεί σε μια κατάσταση συμπίεσης c_i μεγέθους 256 ή 512 bits που χωρίζεται σε οκτώ λέξεις, A, B, C, D, E, F, G και H , των 32 ή 64 bits η κάθε μια. Αρχικά, η κατάσταση αυτή αρχικοποιείται με προκαθορισμένες αρχικές τιμές που είναι διαφορετικές σε κάθε μέλος της οικογένειας συναρτήσεων SHA-2. Η συμπίεση μηνύματος πραγματοποιείται κάθε φορά για ένα μπλοκ δεδομένων 512 ή 1024 bits και με βάση τα δεδομένα επέκτασης μηνύματος αυτού του μπλοκ τροποποιείται η τρέχουσα κατάσταση σε συνολικά 64 ή 80 γύρους, όπως απεικονίζεται στο δεξιό μέρος του Σχήματος 3.11. Στο σχήμα αυτό, το W_b υποδηλώνει μια λέξη 32 ή 64 bits της επέκτασης μηνύματος, το K_b υποδηλώνει μια σταθερά διαφορετική σε κάθε γύρο, το Σ υποδηλώνει μια πρόσθεση με μόντουλο 2^{32} ή 2^{64} , και οι συναρτήσεις Ch , Ma , Σ_0 και Σ_1 δίνονται από τις ακόλουθες εξισώσεις:

$$\begin{aligned}
 Ch(E, F, G) &= (E \wedge F) \oplus ((\neg E) \wedge G) \\
 Ma(A, B, C) &= (A \wedge B) \oplus (A \wedge C) \oplus (B \wedge C) \\
 \Sigma_0(A) &= \begin{cases} (A \ggg_2) \oplus (A \ggg_{13}) \oplus (A \ggg_{22}) & \text{μπλοκ = 512 bits} \\ (A \ggg_{28}) \oplus (A \ggg_{34}) \oplus (A \ggg_{39}) & \text{μπλοκ = 1024 bits} \end{cases} \\
 \Sigma_1(E) &= \begin{cases} (E \ggg_6) \oplus (E \ggg_{11}) \oplus (E \ggg_{25}) & \text{μπλοκ = 512 bits} \\ (E \ggg_{14}) \oplus (E \ggg_{18}) \oplus (E \ggg_{41}) & \text{μπλοκ = 1024 bits} \end{cases}
 \end{aligned}$$

όπου τα σύβιμολα \oplus , \wedge , \neg και \ggg_x υποδηλώνουν τις πράξεις XOR, AND, NOT και δεξιά ολίσθηση κατά x bit, αντίστοιχα.

Τέλος, κατά την ενημέρωση σύνοψης, η οικογένεια συναρτήσεων SHA-2 ορίζει την αρχική κατάσταση σύνοψης h_0 μεγέθους 224, 256, 384 ή 512 bits με βάση προκαθορισμένες αρχικές τιμές (τις ίδιες με την συμπίεση μηνύματος), ενώ μετά το τέλος της συμπίεσης ενός μπλοκ, προσθέτει στην τρέχουσα κατάσταση σύνοψης h_i την κατάσταση συμπίεσης c'_i που προέκυψε από το τρέχον μπλοκ m_i , δηλ. $h_{i+1} = h_i + c'_i$. Η τελική σύνοψη $H(m)$ της οικογένειας συναρτήσεων SHA-2 αποτελείται από την συνένωση των οκτώ λέξεων της τελικής κατάστασης σύνοψης h_n , δηλ. $H(m) = h_n^A \parallel h_n^B \parallel h_n^C \parallel h_n^D \parallel h_n^E \parallel h_n^F \parallel h_n^G \parallel h_n^H$, όπου n το πλήθος των μπλοκ.

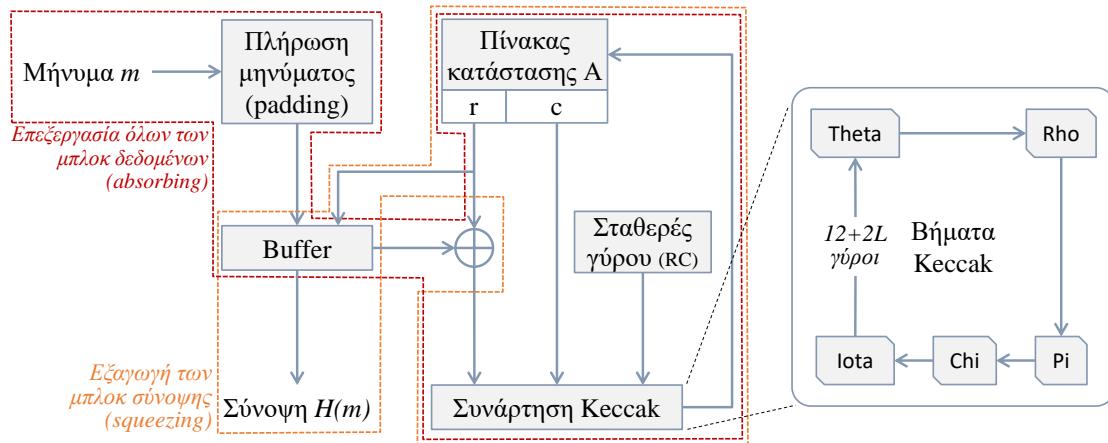
3.5.3 Συνάρτηση Σύνοψης SHA-3

Η συνάρτηση σύνοψης SHA-3 [26] αποτελεί το τελευταίο μέλος της οικογένειας συναρτήσεων σύνοψης SHA και κυκλοφόρησε από το NIST τον Αύγουστο του 2015. Αν και αποτελεί μέρος της ίδιας σειράς συναρτήσεων σύνοψης, η συνάρτηση SHA-3 είναι εσωτερικά αρκετά διαφορετική σε σχέση με τη δομή των SHA-1 και SHA-2 που μοιάζουν με τον MD5. Η συνάρτηση σύνοψης SHA-3 αποτελεί ουσιαστικά ένα υποσύνολο της ευρύτερης οικογένειας συναρτήσεων σύνοψης Keccak [27] που προτάθηκε από τους Bertoni, Daemen, Peeters και Van Assche το 2013. Οι δημιουργοί του Keccak έχουν επιπλέον προτείνει πρόσθετες λειτουργίες τη συνάρτησης σύνοψης, όπως μιας κρυπτογράφησης ροής και ενός σχήματος σύνοψης δεντρικής μορφής, που ωστόσο δεν έχουν ακόμη προτυποποιηθεί από το NIST.

Η συνάρτηση σύνοψης SHA-3 βασίζεται σε μια κατασκευή Sponge (Ενότητα 3.2.4) που επιτρέπει την εισαγωγή («απορρόφηση») οποιασδήποτε ποσότητας δεδομένων και την εξαγωγή («συμπίεση») οποιασδήποτε ποσότητας δεδομένων, ενώ λειτουργεί ως μια ψευδοτυχαία συνάρτηση σε σχέση με όλες τις προηγούμενες εισόδους. Η γενιά συναρτήσεων σύνοψης SHA-3 περιλαμβάνει κατά κύριο λόγο τις συναρτήσεις σύνοψης SHA3-224, SHA3-256, SHA3-384, και SHA3-512, και επιπλέον τις SHAKE128 και SHAKE256 που επιτρέπουν αυθαίρετον μήκους έξodo σύνοψης.

Όπως φαίνεται στο Σχήμα 3.12, η συνάρτηση σύνοψης SHA-3 αποτελείται από τέσσερις κύριες περιοχές: την πλήρωση (padding) μηνύματος, τον πίνακα κατάστασης, το buffer, και την συνάρτηση Keccak. Για την πλήρωση ενός μηνύματος με μέγεθος μπλοκ r bit χρησιμοποιείται το μοτίβο 1 0 ... 1, όπου αρχικά ένα bit με τιμή “1” τοποθετείται στο τέλος των δεδομένων, ακολουθούν k bits με τιμή “0” (με μέγιστο $r - 1$), καθώς και ένα τελευταίο bit με τιμή “1”. Το μέγιστος πλήθος $r - 1$ μηδενικών συμβαίνει όταν το τελευταίο μπλοκ δεδομένων του μηνύματος έχει μήκος $r - 1$ bits. Σε αυτήν την περίπτωση, ένα νέο μπλοκ προστίθεται μετά το αρχικό bit με τιμή “1”, που περιέχει $r - 1$ μηδενικά bit πριν από το τελευταίο bit με τιμή “1”. Ο πίνακας κατάστασης έχει μήκος $b = 1600$ bits και αποτελείται από έναν τρισδιάστατο πίνακα μεγέθους $5 \times 5 \times w$ λέξεων, όπου το μέγεθος της κάθε λέξης μπορεί να είναι $w = 2^L$. Στους αλγορίθμους που έχουν προτυποποιηθεί από το NIST [26], το μέγεθος λέξης που έχει θεσπιστεί είναι 64 bits για $L = 6$. Επιπλέον, ο πίνακας κατάστασης χωρίζεται σε δύο μέρη με r και c bits (όπου $b = r + c$), που αρχικοποιούνται με την τιμή “0”. Τα δύο αυτά μέρη αντιπροσωπεύουν τα εξής:

- Το r αποτελεί την εξωτερική κατάσταση του αλγορίθμου που αντιστοιχεί στο μέγεθος του μπλοκ δε-



Σχήμα 3.12: Λειτουργία της συνάρτησης σύνοψης SHA-3.

δομένων και ονομάζεται ρυθμός (rate).

- Το c αποτελεί την εσωτερική κατάσταση του αλγορίθμου που ορίζει το επίπεδο ασφαλείας και ονομάζεται χωρητικότητα (capacity).

Το buffer, κατά την φάση «απορρόφησης», δέχεται ως είσοδο δύο παραμέτρους: η πρώτη είναι ο αριθμός των bits (64 ή 256 bits) που δέχεται ως δεδομένα εισόδου από το μήνυμα και η δεύτερη είναι ο πίνακας κατάστασης. Αντίστοιχα, κατά την φάση «συμπίεσης», το buffer παράγει ως έξοδο τη τελική σύνοψη $H(m)$.

Η συνάρτηση Keccak εσωτερικά αποτελείται από πέντε βήματα, το Θ , ρ , π , χ , και ι , που αλληλεπιδρούν με τον πίνακα κατάστασης A και τα οποία δίνονται από τις ακόλουθες εξισώσεις:

$$\begin{aligned} \Theta(\theta) : \quad C[x] &= A[x, 0] \oplus A[x, 1] \oplus A[x, 2] \oplus A[x, 3] \oplus A[x, 4] & 0 \leq x \leq 4 \\ &D[x] = C[x - 1] \oplus (C[x + 1] \lll_1) & 0 \leq x \leq 4 \\ &A[x, y] = A[x, y] \oplus D[x] & 0 \leq x, y \leq 4 \end{aligned}$$

$$\rho(\rho) \& \pi(\pi) : \quad B[y, 2x + 3y] = A[x, y] \lll_{r[x,y]} \quad 0 \leq x, y \leq 4$$

$$\chi(\chi) : \quad A[x, y] = B[x, y] \oplus ((\neg B[x + 1, y]) \wedge B[x + 2, y]) \quad 0 \leq x, y \leq 4$$

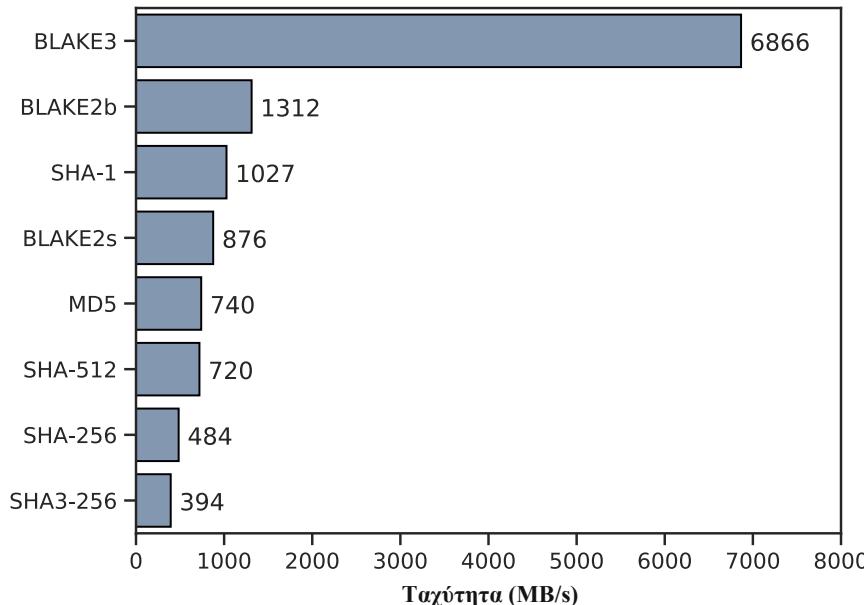
$$\iota(\iota) : \quad A[0, 0] = A[0, 0] \oplus RC$$

Στις παραπάνω εξισώσεις, το Θ αποτελείται από έναν υπολογισμό ισοτιμίας ($C[x]$), μια ολίσθηση (\lll_1) κατά μια θέση και πράξεις XOR (\oplus), το ρ αποτελείται μια ολίσθηση ($\lll_{r[x,y]}$) κατά μια μετατόπιση που εξαρτάται από τη θέση της λέξης, το π πραγματοποιεί μια μετάθεση ($B[y, 2x + 3y]$), το χ αποτελείται από δυαδικές πράξεις XOR (\oplus), NOT (\neg) και AND (\wedge), και το ι πραγματοποιεί την προσθήκη μια σταθεράς (RC) διαφορετικής ανά γύρο. Τα πέντε αυτά βήματα επαναλαμβάνονται για $12 + 2L$ γύρους, όπου σύμφωνα με την προτυποποίηση του NIST [26] αυτό αντιστοιχεί σε 24 γύρους (για $L = 6$).

3.6 Οικογένεια Συναρτήσεων Σύνοψης BLAKE

Η οικογένεια συναρτήσεων σύνοψης BLAKE αποτελεί ένα σύνολο συναρτήσεων σύνοψης που μέχρι στιγμής απαρτίζεται από τρία μέλη, τις συναρτήσεις BLAKE, BLAKE2 και BLAKE3. Το πρώτο μέλος της οικογένειας, ο BLAKE [9], υποβλήθηκε το 2008 στον διαγωνισμό του NIST [28] για την ανάδειξη του νέου αλγορίθμου σύνοψης SHA-3 και ήταν ανάμεσα στους πέντε φιναλίστ μαζί με τον Keccak [27]. Ο BLAKE2

[29] προτάθηκε το 2012 και ουσιαστικά αποτελεί μια βελτιωμένη έκδοση του BLAKE ενισχύοντας αρκετά την ταχύτητά του. Τέλος, ο BLAKE3 [17] προτάθηκε το 2020, βελτιώνοντας ακόμη περισσότερο την ταχύτητα των προκατόχων του (Σχήμα 3.13).



Σχήμα 3.13: Συγκριτική παρουσίαση ταχύτητας της οικογένεια συναρτήσεων BLAKE για δεδομένα εισόδου 16 KB σε έναν σύγχρονο εξυπηρετητή (επεξεργαστής Cascade Lake-SP 8275CL).

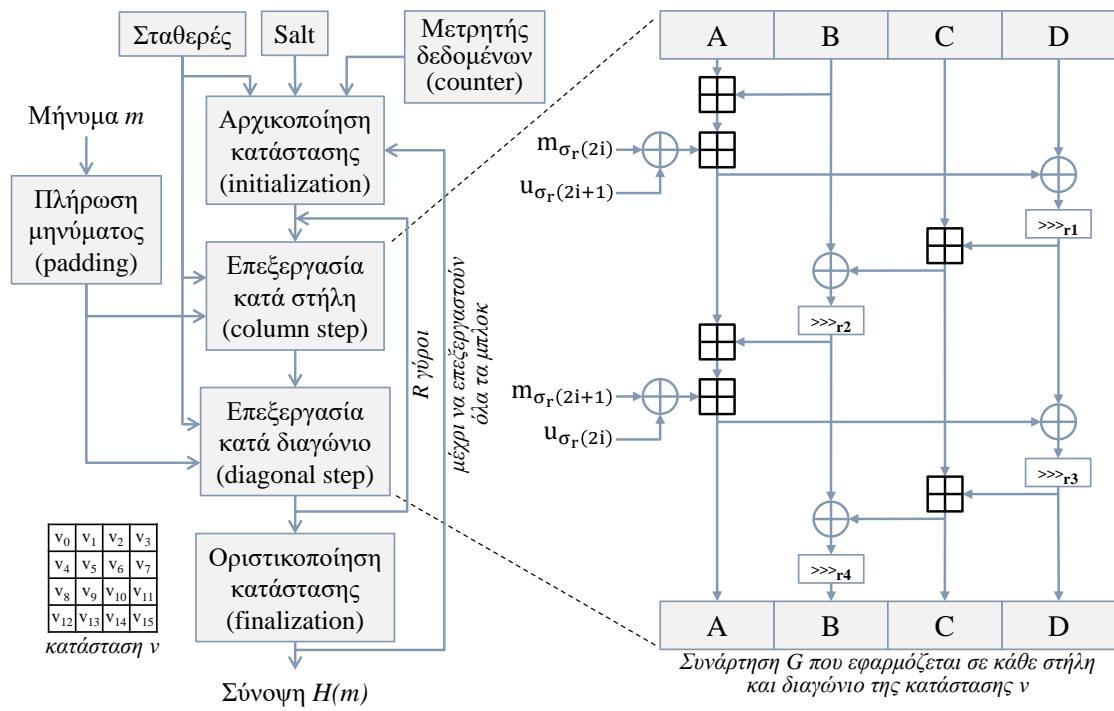
3.6.1 Συνάρτηση Σύνοψης BLAKE

Η συνάρτηση σύνοψης BLAKE [9] περιλαμβάνει κατά κύριο λόγο τις συναρτήσεις BLAKE-224, BLAKE-256, BLAKE-384 και BLAKE-512, των οποίων τα χαρακτηριστικά εμφανίζονται στον Πίνακα 3.2. Όπως και με την συνάρτηση σύνοψης SHA-2, η συνάρτηση BLAKE παρέχεται με μια 32-bit έκδοση (BLAKE-256) και μια 64-bit έκδοση (BLAKE-512), από τις οποίες οι άλλες παραλλαγές του (BLAKE-224 και BLAKE-384) προέρχονται μέσω τροποποίησης των παραμέτρων.

Πίνακας 3.2: Συγκριτική παρουσίαση της οικογένειας συναρτήσεων σύνοψης BLAKE.

Όνομα Αλγορίθμου	Μέγεθος Εξόδου (bits)	Μέγεθος Εσωτερικής Κατάστασης (bits)	Μέγεθος Μπλοκ Εισόδου (bits)	Salt (bits)	Γύροι
BLAKE-224	224	512	512	128	14
BLAKE-256	256	(4 x 4 x 32)			
BLAKE-384	384	1024	1024	256	16
BLAKE-512	512	(4 x 4 x 64)			

Ηλειτουργία της συνάρτησης σύνοψης BLAKE βασίζεται στην κατασκευή HAIFA (Ενότητα 3.2.3) και σε υψηλό επίπεδο μπορεί να χωριστεί σε πέντε κύριες περιοχές: την πλήρωση (padding) μηνύματος, την αρχικοποίηση κατάστασης, την επεξεργασία κατά στήλη (column step), την επεξεργασία κατά διαγώνιο (diagonal step), και την οριστικοποίηση κατάστασης. Στο Σχήμα 3.14 αποτυπώνεται η περιγραφή της λειτουργίας της συνάρτησης σύνοψης BLAKE και ακολουθεί η αναλυτική περιγραφή της.



Σχήμα 3.14: Λειτουργία της συνάρτησης σύνοψης BLAKE.

Αρχικά, κατά την πλήρωση μηνύματος, τα δεδομένα του μηνύματος συμπληρώνονται κατάλληλα στο τέλος έτσι ώστε το μέγεθος τους να γίνει πολλαπλάσιο των 512 ή 1024 bits, ανάλογα με την έκδοση της συνάρτησης BLAKE. Η διαδικασία πλήρωσης γίνεται ως εξής:

1. Προστίθεται στα δεδομένα ένα bit με τιμή “1” ακολουθούμενο από έναν ελάχιστο αριθμό από bit με τιμή “0” έτσι ώστε το συνολικό μέγεθος να είναι σύμφωνο με $447 \bmod 512 \text{ ή } 895 \bmod 1024$. Έτσι, προστίθεται τουλάχιστον ένα bit και το πολύ ακόμη 512 ή 1024.
2. Επιπλέον, προστίθεται ένα bit με τιμή “1” ακολουθούμενο από το μήκος του μηνύματος πριν την πλήρωσή του, ως ένας unsigned big-endian μεγέθους 64 ή 128 bits. Εδώ, θα πρέπει να σημειωθεί ότι στις εκδόσεις BLAKE-224 και BLAKE-384, το bit που προστίθεται έχει την τιμή “0” και όχι “1”.

Η εσωτερική κατάσταση της συνάρτησης συμπίεσης έχει μέγεθος 512 ή 1024 bits και αναπαρίσταται με ένα πίνακα 4×4 , ο οποίος διαθέτει λέξεις των 32 ή 64 bits, αντίστοιχα. Η αρχικοποίηση της εσωτερικής κατάστασης γίνεται ως εξής:

$$\begin{pmatrix} v_0 & v_1 & v_2 & v_3 \\ v_4 & v_5 & v_6 & v_7 \\ v_8 & v_9 & v_{10} & v_{11} \\ v_{12} & v_{13} & v_{14} & v_{15} \end{pmatrix} = \begin{pmatrix} h_0 & h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 & h_7 \\ s_0 \oplus u_0 & s_1 \oplus u_1 & s_2 \oplus u_2 & s_3 \oplus u_3 \\ t_0 \oplus u_4 & t_1 \oplus u_5 & t_2 \oplus u_6 & t_3 \oplus u_7 \end{pmatrix}$$

όπου το $h = h_0 \parallel \dots \parallel h_7$ αποτελεί την τρέχουσα τιμή σύνοψης η οποία στο πρώτο μπλοκ δεδομένων αρχικοποιείται με προκαθορισμένες αρχικές τιμές (IV), το $s = s_0 \parallel s_1 \parallel s_3 \parallel s_4$ αποτελεί το «αλάτι» (salt) μεγέθους 128 ή 256 bits (ανάλογα με την έκδοση), το $t = t_0 \parallel t_1$ είναι ο μετρητής μεγέθους 128 bits, και το u αντιπροσωπεύει 16 προκαθορισμένες σταθερές.

Κατά την επεξεργασία κατά στήλη και διαγώνιο του πίνακα της εσωτερικής κατάστασης, πραγματοποιούνται 14 ή 16 γύροι (ανάλογα με την έκδοση) εφαρμόζοντας κατάλληλους μετασχηματισμούς με χρήση της συνάρτησης G που απεικονίζεται στο δεξιό μέρος του Σχήματος 3.14. Στο σχήμα αυτό, το $\sigma_r(2i)$ ή $\sigma_r(2i + 1)$

αποτελεί έναν προκαθορισμένο πίνακα μεταθέσεων 10×16 (όπου $r = R \bmod 10$ για τον τρέχοντα γύρο R και i ο μετασχηματισμούς με την συνάρτηση G), το m_{σ_r} υποδηλώνει μια λέξη 32 ή 64 bits από το μπλοκ του μηνύματος, το u_{σ_r} υποδηλώνει μια σταθερά 32 ή 64 bits, τα $\ggg_{r1}, \ggg_{r2}, \ggg_{r3}$ και \ggg_{r4} υποδηλώνουν μια δεξιά ολίσθηση κατά 16, 12, 8 και 7 bit στις συναρτήσεις BLAKE-256/224 και κατά 32, 25, 16 και 11 bit στις συναρτήσεις BLAKE-512/384, το \oplus υποδηλώνει μια πρόσθεση με μόντουλο 2^{32} ή 2^{64} , και το \oplus υποδηλώνει μια πράξη XOR. Οι μετασχηματισμοί που πραγματοποιούνται σε κάθε γύρο με βάση την συνάρτηση G_i είναι οι εξής:

Κατά στήλη:

$$G_0(v_0, v_4, v_8, v_{12}) \quad G_1(v_1, v_5, v_9, v_{13}) \quad G_2(v_2, v_6, v_{10}, v_{14}) \quad G_3(v_3, v_7, v_{11}, v_{15})$$

Κατά διαγώνιο:

$$G_4(v_0, v_5, v_{10}, v_{15}) \quad G_5(v_1, v_6, v_{11}, v_{12}) \quad G_6(v_2, v_7, v_8, v_{13}) \quad G_7(v_3, v_4, v_9, v_{14})$$

Μετά τη ολοκλήρωση των γύρων, πραγματοποιείται η οριστικοποίηση κατάστασης με παραγωγή της νέας τιμής σύνοψης $h' = h'_0 \parallel \dots \parallel h'_7$ που ορίζεται ως συνδυασμός της τελικής τιμής του πίνακα εσωτερικής κατάστασης v με την προηγούμενη τιμή σύνοψης h και το salt s :

$$\begin{aligned} h'_0 &= h_0 \oplus s_0 \oplus v_0 \oplus v_8 \\ h'_1 &= h_1 \oplus s_1 \oplus v_1 \oplus v_9 \\ h'_2 &= h_2 \oplus s_2 \oplus v_2 \oplus v_{10} \\ h'_3 &= h_3 \oplus s_3 \oplus v_3 \oplus v_{11} \\ h'_4 &= h_4 \oplus s_0 \oplus v_4 \oplus v_{12} \\ h'_5 &= h_5 \oplus s_1 \oplus v_5 \oplus v_{13} \\ h'_6 &= h_6 \oplus s_2 \oplus v_6 \oplus v_{14} \\ h'_7 &= h_7 \oplus s_3 \oplus v_7 \oplus v_{15} \end{aligned}$$

Σε αυτό το σημείο θα πρέπει να επισημανθεί ότι στην περίπτωση του BLAKE-224 η τιμή σύνοψης h αποτελείται από τα πρώτα 224 bits (7×32 bits), δηλ. $h_0 \parallel \dots \parallel h_6$, και στον BLAKE-384 αποτελείται από τα πρώτα 384 bits (6×64 bits), δηλ. $h_0 \parallel \dots \parallel h_5$.

3.6.2 Συνάρτηση Σύνοψης BLAKE2

Η συνάρτηση σύνοψης BLAKE2 [29] αποτελεί απόγονο της συνάρτησης BLAKE και είχε ως στόχο την αντικατάσταση των ευρέως χρησιμοποιούμενων συναρτήσεων MD5 και SHA-1 σε εφαρμογές που απαιτούν υψηλή απόδοση, παρέχοντας ταυτόχρονα καλύτερη ασφάλεια από την συνάρτηση σύνοψης SHA-2 και παρόμοια με αυτή της συνάρτηση σύνοψης SHA-3. Η συνάρτηση σύνοψης BLAKE2 αποτελείται από δύο κύριες εκδόσεις:

- Η συνάρτηση BLAKE2b είναι βελτιστοποιημένος για πλατφόρμες 64 bits και παράγει συνόψεις οποιουδήποτε μεγέθους μεταξύ 1 και 64 bytes, με χαρακτηριστικά παραδείγματα τις συναρτήσεις BLAKE2b-512 και BLAKE2b-384.
- Η συνάρτηση BLAKE2s είναι βελτιστοποιημένος για πλατφόρμες 8 έως 32 bits και παράγει συνόψεις οποιουδήποτε μεγέθους μεταξύ 1 και 32 bytes, με χαρακτηριστικά παραδείγματα τις συναρτήσεις BLAKE2s-256 και BLAKE2s-224.

Οι κύριες διαφορές της συνάρτησης σύνοψης BLAKE2 σε σχέση με την συνάρτηση BLAKE είναι οι εξής:

- Διαθέτει μικρότερο αριθμό γύρων. Η συνάρτηση BLAKE2b έχει 12 γύρους και η συνάρτηση BLAKE2s έχει 10 γύρους, έναντι 16 και 14, αντίστοιχα.

- Διαθέτει βελτιστοποίηση των bit ολισθήσεως για λόγους ταχύτητας στην συνάρτηση BLAKE2b από 32, 25, 16 και 11 θέσεις, σε 32, 24, 16 και 63 θέσεις. Αντίστοιχα, η συνάρτηση BLAKE2s διαθέτει τις ίδιες ακριβώς ολισθήσεις (δηλ., 16, 12, 8 και 7) με την συνάρτηση BLAKE.
- Η πλήρωση μηνύματος γίνεται με απλά μηδενικά στο τέλος του μηνύματος και μόνο εάν απαιτείται στο τελευταίο μπλοκ.
- Απαιτεί μικρότερο αριθμό σταθερών, χρησιμοποιώντας μόνο 8 σταθερές αρχικοποίησης (IV), έναντι 24 σταθερών ($8 \text{ IV} + 16 u$) της συνάρτησης BLAKE.
- Ο μετρητής t μετράει bytes και όχι bits.

3.6.3 Συνάρτηση Σύνοψης BLAKE3

Η συνάρτηση σύνοψης BLAKE3 [17] αποτελεί μια αρκετά σύγχρονη και αποδοτική συνάρτηση σύνοψης που βασίζεται στην συνάρτηση σύνοψης BLAKE2. Σε αντίθεση με τις συναρτήσεις BLAKE και BLAKE2, η συνάρτηση BLAKE3 διαθέτει μια και μόνο έκδοση (256 bits εξόδου) με αρκετά επιθυμητά χαρακτηριστικά, όπως παραλληλισμό και επεκτάσιμο μέγεθος εξόδου (XOF). Μια επίσης αρκετά σημαντική διαφορά της συνάρτησης BLAKE3 είναι ότι η λειτουργία της βασίζεται σε μια δυαδική δομή δέντρου Merkle (Ενότητα 3.2.6), επομένως υποστηρίζει πρακτικά απεριόριστο βαθμό παραλληλισμού για μια επαρκή είσοδο δεδομένων. Η συνάρτηση BLAKE3 έχει σχεδιαστεί για να είναι όσο το δυνατόν πιο γρήγορη. Είναι μάλιστα αρκετές φορές πιο γρήγορη από την συνάρτηση BLAKE2, όπως φαίνεται άλλωστε και στο Σχήμα 3.13. Η συνάρτηση σύνοψης BLAKE3 βασίζεται στενά σε αυτή της συνάρτησης BLAKE2s, με τη μεγαλύτερη διαφορά της να αφορά τον αριθμό των γύρων που μειώνεται από 10 σε 7, μια αλλαγή που βασίζεται στην υπόθεση ότι η τρέχουσα κρυπτογραφία είναι υπερβολικά συντηρητική [30]. Εκτός από την παροχή παραλληλισμού, η μορφή δέντρου Merkle επιτρέπει επίσης πολύ γρήγορη επαλήθευση (on-the-fly) και σταδιακές ενημερώσεις των δεδομένων προς σύνοψη.

3.7 Ασκήσεις-Εργασίες

Ασκήσεις

3.7.1 Λαμβάνοντας υπόψη την ακόλουθη είσοδο (4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199) και τη συνάρτηση σύνοψης $h(x) = x \bmod 10$, ποιες από τις παρακάτω προτάσεις είναι αληθείς;

- Οι συνόψεις των 3 μηνυμάτων 9679, 1989 και 4199 είναι ίδιες.
- Οι τιμές 1471, 6171 έχουν την ίδια τιμή σύνοψης.
- Όλες οι τιμές εισόδου έχουν την ίδια τιμή σύνοψης.
- Κάθε τιμή εισόδου έχει μια διαφορετική τιμή σύνοψης.

3.7.2 Μια συνάρτηση σύνοψης h λαμβάνει ως είσοδο ένα μήνυμα αυθαίρετου μήκους και παράγει ως έξοδο μια σύνοψη μηνύματος σταθερού μήκους, για παράδειγμα 160 bit. Ωστόσο, ορισμένες επιπλέον ιδιότητες πρέπει να ικανοποιούνται:

- Με δεδομένο ένα μήνυμα m , η σύνοψη του μηνύματος $h(m)$ μπορεί να υπολογίζεται πολύ γρήγορα.
- Λαμβάνοντας υπόψη μια σύνοψη μηνύματος y , είναι υπολογιστικά ανέφικτο να βρεθεί ένα μήνυμα m με $h(m) = y$ (με άλλα λόγια, το h είναι μια μονόδρομη συνάρτηση).
- Είναι υπολογιστικά ανέφικτο να βρεθούν δύο μηνύματα m_1 και m_2 με $h(m_1) = h(m_2)$ (στην περίπτωση αυτή, η συνάρτηση h λέγεται ότι είναι ανθεκτική σε συγκρούσεις (collision resistant)).

Έστω ότι το n ένας μεγάλος ακέραιος αριθμός και η συνάρτηση σύνοψης $h(m) = m \pmod n$ έχει ως έξοδος συνόψεις που είναι ακέραιοι αριθμοί από το 0 έως και το $n - 1$. Δείξτε ότι η συνάρτηση σύνοψης h ικανοποιεί το (α), αλλά όχι τα (β) και (γ).

3.7.3 Δίνονται δύο πρωτόκολλα στα οποία η πλευρά του αποστολέα εκτελεί τις ακόλουθες πράξεις:

Πρωτόκολλο Α':

$$y = E_{k_1}(x \parallel H(k_2 \parallel x))$$

όπου x είναι το μήνυμα, H είναι μια συνάρτηση σύνοψης όπως η SHA-1, E είναι ένας αλγόριθμος κρυπτογράφησης συμμετρικού κλειδιού, το σύμβολο \parallel υποδηλώνει μια απλή αλληλουχία, και τα k_1, k_2 είναι δύο μυστικά κλειδιά που είναι γνωστά μόνο στον αποστολέα και στον παραλήπτη.

Πρωτόκολλο Β':

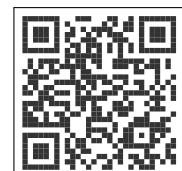
$$y = x \parallel E_{p_k}(H(x))$$

όπου E είναι ένας αλγόριθμος κρυπτογράφησης δημοσίου κλειδιού, p_k είναι το δημόσιο κλειδί του παραλήπτη το οποίο είναι δημόσια γνωστό σε όλους, και το s_k είναι το αντίστοιχο ιδιωτικό κλειδί του παραλήπτη (δεν είναι γνωστό στον αποστολέα).

- (1) Περιγράψτε βήμα προς βήμα (π.χ. με μια αναλυτική λίστα) το τι κάνει ο παραλήπτης μετά τη λήψη του y .
- (2) Εξηγήστε για κάθε μια από τις υπηρεσίες ασφαλείας, εμπιστευτικότητα, ακεραιότητα, και μη-αποποίηση (αποτροπή μιας οντότητας από το να αρνηθεί προηγούμενες δεσμεύσεις ή ενέργειες), εάν διασφαλίζονται ή όχι για καθένα από τα δύο πρωτόκολλα που δίνονται.

Εργασίες

3.7.1 Σε αυτήν την εργασία θα εξετάσετε την περίπτωση όπου θέλουμε να παράγουμε την έξοδο από διάφορες συναρτήσεις σύνοψης κάνοντας χρήση μιας δεδομένης εισόδου κειμένου. Συγκεκριμένα, να χρησιμοποιήσετε τις ακόλουθες έξι κρυπτογραφικές συναρτήσεις σύνοψης: MD5, Whirlpool, SHA-1, SHA-256, SHA3-256, και BLAKE-512 (με χρήση salt). Ως είσοδο κειμένου, καλείστε να χρησιμοποιήσετε το μήνυμα “Hello to the world of cryptographic hashes!” και για την συνάρτηση σύνοψης που χρειάζεται salt να χρησιμοποιήσετε το μήνυμα “This is a salt for BLAKE hashes.” (32 χαρακτήρων ή 256 bits). Στην εργασία αυτή να κάνετε χρήση του εργαλείου [CrypTool 2](#).



3.7.2 Σε αυτήν την εργασία θα δοκιμάσετε διάφορες κρυπτογραφικές συναρτήσεις σύνοψης με χρήση της γλώσσας προγραμματισμού [Java](#) και το περιβάλλον ανάπτυξης [Eclipse IDE for Java Developers](#). Πιο αναλυτικά, θα δοκιμάσετε τους αλγορίθμους συναρτήσεις σύνοψης MD5, SHA-1, SHA-256 (με και χωρίς salt), SHA-384, SHA-512, [Bcrypt](#) και [Scrypt](#). Κάνοντας χρήση του Eclipse Project “[crypto_chap03](#)”, οι δοκιμές που θα πρέπει να γίνουν είναι οι εξής:



- (1) Για να δοκιμάσετε μεμονωμένα την συνάρτηση σύνοψης MD5 εκτελέστε το αρχείο `TestMD5.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να αλλάξετε το μήνυμα από “Cryptography” σε “cryptography” και δείτε ποιο είναι το αποτέλεσμα σύνοψης. Παρουσιάζει κάποια ομοιότητα με την προηγούμενη σύνοψη;
- (2) Για να δοκιμάσετε μεμονωμένα την συνάρτηση σύνοψης SHA-1 εκτελέστε το αρχείο `TestSHA1.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να προσθέσετε ως salt το “hello” πριν την προσθήκη του μηνύματος (δηλ., “`s.update("hello".getBytes());`”) και δείτε

ποιο είναι το αποτέλεσμα σύνοψης. Παρουσιάζει κάποια ομοιότητα με την προηγούμενη σύνοψη; Για ποιον λόγο γίνεται χρήση salt στις συναρτήσεις σύνοψης;

- (3) Για να δοκιμάσετε τις υπόλοιπες συναρτήσεις σύνοψης SHA-256 (με και χωρίς salt), SHA-384, SHA-512, Bcrypt και Scrypt εκτελέστε το αρχείο `TestHashAlgs.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να αλλάξετε το μήνυμα από “Cryptography” σε κάποιο άλλο της επιλογής σας και παρατηρήστε τις διάφορες συνόψεις που παράγονται.

Βιβλιογραφία

- [1] Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 2018.
- [2] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. 20th Anniversary. John Wiley & Sons, 2015. ISBN: 978-1-119-09672-6.
- [3] George Drosatos. “Utilization and Protection of Personal Data in Ubiquitous Computing Environments”. English. PhD thesis. University Campus, Xanthi 67100, Greece: Department of Electrical and Computer Engineering, Democritus University of Thrace, July 2013. doi: 10.12681/eadd/30085.
- [4] Kathleen Moriarty, Burt Kaliski, Jakob Jonsson, and Andreas Rusch. *PKCS #1: RSA Cryptography Specifications Version 2.2*. RFC 8017. <https://www.rfc-editor.org/rfc/rfc8017.txt>. RFC Editor, Nov. 2016.
- [5] Seny Kamara. “Proofs of Storage: Theory, Constructions and Applications”. In: *Algebraic Informatics*. Ed. by Traian Muntean, Dimitrios Poulakis, and Robert Rolland. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 7–8. ISBN: 978-3-642-40663-8. doi: 10.1007/978-3-642-40663-8_4.
- [6] Pouria Pirzadeh, Junichi Tatenuma, Oliver Po, and Hakan Hacigümüş. “Performance evaluation of range queries in key value stores”. In: *Journal of Grid Computing* 10.1 (2012), pp. 109–132. doi: 10.1007/s10723-012-9214-7.
- [7] Penny Pritzker and Willie E. May. “Secure Hash Standard”. In: *Federal Information Processing Standards Publication FIPS PUB 180-4* (2015), pp. 1–31. doi: 10.6028/NIST.FIPS.180-4.
- [8] Stefan Lucks. “A Failure-Friendly Design Principle for Hash Functions”. In: *Advances in Cryptology - ASIACRYPT 2005*. Ed. by Bimal Roy. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 474–494. ISBN: 978-3-540-32267-2. doi: 10.1007/11593447_26.
- [9] Jean-Philippe Aumasson, Willi Meier, Raphael C.-W. Phan, and Luca Henzen. *The Hash Function BLAKE*. Springer, 2014. ISBN: 978-3-662-44756-7. doi: 10.1007/978-3-662-44757-4.
- [10] Eli Biham and Orr Dunkelman. *A Framework for Iterative Hash Functions – HAIFA*. Cryptology ePrint Archive, Report 2007/278. <https://ia.cr/2007/278>. 2007.
- [11] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. “On the Indifferentiability of the Sponge Construction”. In: *Advances in Cryptology – EUROCRYPT 2008*. Ed. by Nigel Smart. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 181–197. ISBN: 978-3-540-78967-3. doi: 10.1007/978-3-540-78967-3_11.
- [12] Penny Pritzker and Willie May. “SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions”. In: *Federal Information Processing Standards Publication FIPS PUB 202* (2015), pp. 1–29. doi: 10.6028/NIST.FIPS.202.

- [13] Phillip Rogaway and John Steinberger. "Security/Efficiency Tradeoffs for Permutation-Based Hashing". In: *Advances in Cryptology – EUROCRYPT 2008*. Ed. by Nigel Smart. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 220–236. ISBN: 978-3-540-78967-3. DOI: 10.1007/978-3-540-78967-3_13.
- [14] Bart Preneel, René Govaerts, and Joos Vandewalle. "Hash functions based on block ciphers: a synthetic approach". In: *Advances in Cryptology — CRYPTO' 93*. Ed. by Douglas R. Stinson. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 368–378. ISBN: 978-3-540-48329-8. DOI: 10.1007/3-540-48329-2_31.
- [15] Ralph C. Merkle. "A Digital Signature Based on a Conventional Encryption Function". In: *Advances in Cryptology — CRYPTO '87*. Ed. by Carl Pomerance. Berlin, Heidelberg: Springer Berlin Heidelberg, 1988, pp. 369–378. ISBN: 978-3-540-48184-3. DOI: 10.1007/3-540-48184-2_32.
- [16] Leslie Lamport. "Password Authentication with Insecure Communication". In: *Communications of the ACM* 24.11 (Nov. 1981), pp. 770–772. ISSN: 0001-0782. DOI: 10.1145/358790.358797.
- [17] Jack O'Connor, Jean-Philippe Aumasson, Samuel Neves, and Zooko Wilcox-O'Hearn. *BLAKE3: One Function, Fast Everywhere*. <https://blake3.io>. 2020.
- [18] Ronald Rivest. *The MD5 Message-Digest Algorithm*. RFC 1321. <https://www.rfc-editor.org/rfc/rfc1321.txt>. RFC Editor, Apr. 1992. DOI: 10.17487/RFC1321.
- [19] Tao Xie, Fanbao Liu, and Dengguo Feng. *Fast Collision Attack on MD5*. Cryptology ePrint Archive, Report 2013/170. <https://ia.cr/2013/170>. 2013.
- [20] William Stallings. "The Whirlpool Secure Hash Function". In: *Cryptologia* 30.1 (2006), pp. 55–67. DOI: 10.1080/01611190500380090.
- [21] NESSIE Project. *New European Schemes for Signatures, Integrity, and Encryption*. IST-1999-12324. <https://www.cosic.esat.kuleuven.be/nessie/>. Mar. 2000.
- [22] *IT Security techniques – Hash-functions – Part 3: Dedicated hash-functions*. Standard. International Organization for Standardization/International Electrotechnical Commission, 2006, pp. 1–398.
- [23] Ronald H. Brown and Raymond G. Kammer. "Secure Hash Standard". In: *Federal Information Processing Standards Publication FIPS PUB 180* (1993), pp. 1–20. DOI: 10.6028/NIST.FIPS.180.
- [24] Ronald H. Brown, Mary L. Good, and Arati Prabhakar. "Secure Hash Standard". In: *Federal Information Processing Standards Publication FIPS PUB 180-1* (1995), pp. 1–21. DOI: 10.6028/NIST.FIPS.180-1.
- [25] Carlos M. Gutierrez and Patrick Gallagher. "Secure Hash Standard". In: *Federal Information Processing Standards Publication FIPS PUB 180-3* (2008), pp. 1–27.
- [26] Penny Pritzker and Willie May. "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions". In: *Federal Information Processing Standards Publication FIPS PUB 202* (2015), pp. 1–29. DOI: 10.6028/NIST.FIPS.202.
- [27] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. "Keccak". In: *Advances in Cryptology – EUROCRYPT 2013*. Ed. by Thomas Johansson and Phong Q. Nguyen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 313–314. ISBN: 978-3-642-38348-9. DOI: 10.1007/978-3-642-38348-9_19.
- [28] Richard F. Kayser. "Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHA-3) Family". In: *Federal Register* 72.212 (2007), pp. 62212–62220.

- [29] Jean-Philippe Aumasson, Samuel Neves, Zooko Wilcox-O’Hearn, and Christian Winnerlein. “BLAKE2: Simpler, Smaller, Fast as MD5”. In: *Applied Cryptography and Network Security*. Ed. by Michael Jacobson, Michael Locasto, Payman Mohassel, and Reihaneh Safavi-Naini. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 119–135. ISBN: 978-3-642-38980-1. doi: 10.1007/978-3-642-38980-1_8.
- [30] Jean-Philippe Aumasson. *Too Much Crypto*. Cryptology ePrint Archive, Report 2019/1492. <https://ia.cr/2019/1492>. 2019.

Μέρος II

ΜΗΧΑΝΙΣΜΟΙ ΚΑΙ ΠΡΩΤΟΚΟΛΛΑ

ΚΕΦΑΛΑΙΟ 4

ΓΕΝΝΗΤΡΙΕΣ (ΨΕΥΔΟ)ΤΥΧΑΙΩΝ ΑΡΙΘΜΩΝ

Περίληψη

Με τον όρο γεννήτρια τυχαίων αριθμών [1] εννοείται κάθε υπολογιστική ή φυσική διάταξη, ικανή να παράγει μια ακολουθία αριθμών οι οποίοι δεν μπορούν να προβλεφθούν. Οι τυχαίοι αριθμοί είναι απαραίτητοι σε διάφορα κρυπτογραφικά συστήματα και πρωτόκολλα και αποτελούν αναπόσπαστο κομμάτι της κρυπτογραφίας. Υπάρχουν τρεις μηχανισμοί υπεύθυνοι για την τυχαία συμπεριφορά των συστημάτων: η τυχαιότητα που οφείλεται στις αρχικές συνθήκες, η τυχαιότητα που δημιουργείται από το ίδιο το σύστημα και, τέλος, η τυχαιότητα που οφείλεται στο περιβάλλον. Κάθε ένας από αυτούς τους μηχανισμούς παράγει διαφορετικά επίπεδα τυχαιότητας και κατ’ επέκταση ασφάλειας που μπορεί να παρέχει το εκάστοτε κρυπτογραφικό σύστημα που τον χρησιμοποιεί. Δύο είναι τα κύρια είδη γεννητριών τυχαίων αριθμών: οι γεννήτριες που βασίζονται σε υλικό ειδικού σκοπού και έχουν τη δυνατότητα δημιουργίας πραγματικά τυχαίων αριθμών (True Random Number Generator – TRNG) και αυτές που βασίζονται σε λογισμικό και δημιουργούν τους λεγόμενους ψευδοτυχαίους αριθμούς (Pseudo-Random Number Generator – PRNG). Στο κεφάλαιο αυτό επικεντρωνόμαστε σε γεννήτριες ψευδοτυχαίων αριθμών που συνήθως υλοποιούνται σε επίπεδο λογισμικού. Πιο αναλυτικά, στην Ενότητα 4.1 γίνεται μια εισαγωγή στους τυχαίους αριθμούς, ενώ στην Ενότητα 4.2 ορίζεται η ορολογία που χρησιμοποιείται στο χώρο. Στις Ενότητες 4.3, 4.4 και 4.5 παρουσιάζονται η γενική αρχιτεκτονική των PRNG, οι διάφορες απαιτήσεις ασφαλείας που πρέπει να πληρούνται, καθώς και διάφορα ζητήματα που πρέπει να αντιμετωπιστούν κατά την υλοποίηση των PRNG, αντίστοιχα. Επιπρόσθετα, στην Ενότητα 4.6 γίνεται μια αναλυτική παρουσίαση των μηχανισμών που βασίζονται σε συναρτήσεις σύνοψης (HASH-DRBG), αυτών που βασίζονται σε HMAC (HMAC-DRBG) και εκείνων που βασίζονται σε αλγορίθμους κρυπτογραφίας μπλοκ (CTR-DRBG). Τέλος, στην Ενότητα 4.7 αναλύονται διάφορες εναλλακτικές κρυπτογραφικά ασφαλείς γεννήτριες ψευδοτυχαίων αριθμών, όπως οι αλγόριθμοι Blum-Blum-Shub, Blum-Micali και Yarrow, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών αρχών της συμμετρικής κρυπτογράφησης (Κεφάλαιο 1) και των συναρτήσεων σύνοψης (Κεφάλαιο 3).

4.1 Τυχαίοι Αριθμοί

Οι τυχαίοι αριθμοί αποτελούν ένα βασικό δομικό στοιχείο της κρυπτογραφίας, το οποίο απαιτείται σε όλα σχέδιόν τα κρυπτογραφικά συστήματα και πρωτόκολλα. Για παράδειγμα, τυχαίοι αριθμοί χρειάζονται για τη δημιουργία ζεύγους κλειδιών στην κρυπτογραφία δημοσίου κλειδιού, για τη δημιουργία κλειδιών συμμετρικής κρυπτογραφίας, για τη δημιουργία διανυσμάτων αρχικοποίησης (Initialisation Vector – IV) στους διάφορους τρόπους λειτουργίας των αλγορίθμων κρυπτογράφησης μπλοκ, σε πρωτόκολλα πρόκλησης-απόκρισης (challenge-response), ως πρόσθετες είσοδοι στους περισσότερους αλγόριθμους κρυπτογράφησης και υπογραφής δημόσιου κλειδιού και για τη δημιουργία εφήμερων τιμών στα πρωτόκολλα ανταλλαγής κλειδιών (key exchange). Η ύπαρξη κατάλληλων τυχαίων τιμών θεωρείται δεδομένη σε μεγάλο μέρος της ερευνητικής βιβλιογραφίας στην κρυπτογραφία και σχεδόν όλες οι επίσημες αναλύσεις ασφάλειας κρυπτογραφικών σχημάτων αποτυγχάνουν εάν δεν τηρηθεί η παραδοχή της τέλειας τυχαιότητας (perfect randomness). Ωστόσο, υπάρχουν πολλά εμφανή παραδείγματα αστοχιών στη δημιουργία τυχαίων αριθμών, με σοβαρές συνέπειες στην ασφάλεια. Μερικά αρκετά γνωστά παραδείγματα τέτοιων αστοχιών είναι τα εξής:

- Η υλοποίηση του πρωτοκόλλου SSL από τη Netscape, αποκαλύφθηκε το 1996 [2] ότι χρησιμοποιούσε μια γεννήτρια τυχαίων αριθμών που είχε ως μόνες πηγές εντροπίας για την δημιουργία του σπόρου τυχαιότητας (seed) της γεννήτριας, την ώρα της ημέρας, το αναγνωριστικό διεργασίας και το γονικό αναγνωριστικό διεργασίας.
- Μια εσφαλμένη διορθωτική ενημέρωση (patch) του OpenSSL από έναν προγραμματιστή του λειτουργικού Debian, οδήγησε σε σημαντικά μειωμένη εντροπία στη δημιουργία κλειδιών με το OpenSSL [3]. Η ενημέρωση επηρέασε την δημιουργία κλειδιών SSH, OpenVPN, και DNSSEC, κρυπτογραφικών κλειδιών για χρήση σε πιστοποιητικά X.509 καθώς και κλειδιών συνεδρίας (session keys) που χρησιμοποιούνται σε συνδέσεις SSL/TLS. Όλα τα κλειδιά που παράχθηκαν από τον Σεπτέμβριο του 2006 έως και τον Μάιο του 2008 ήταν πιθανώς ευάλωτα σε επιθέσεις.
- Δύο ανεξάρτητες μελέτες ανάλυσης δημοσίων κλειδιών που υπάρχουν διαθέσιμες στο Διαδίκτυο [4, 5], αποκαλύψαν το 2012, μεταξύ άλλων, ότι πολλά ζεύγη δημόσιων κλειδιών RSA είχαν κοινούς παράγοντες, καθιστώντας την εξαγωγή των αντίστοιχων ιδιωτικών κλειδιών μια σχετικά εύκολη υπόθεση. Τα ζητήματα που εντοπίστηκαν αποδίδονται εν μέρει σε εσφαλμένες διαδικασίες δημιουργίας τυχαίων αριθμών στον πυρήνα του Linux [4].
- Οι Ristenpart και Yilek μελέτησαν πως αντιμετωπίζεται η τυχαιότητα στις επανεκκινήσεις εικονικών μηχανών (virtual machines) [6], ανακαλύπτοντας ότι η κατάσταση μιας γεννήτριας ψευδοτυχαίων αριθμών (Pseudo-Random Number Generator – PRNG) μπορεί συχνά να προβλεφθεί, προκαλώντας σοβαρά ζητήματα ασφάλειας στο πρωτόκολλο TLS, λόγω πιθανής επίθεσης αποκάλυψης του ιδιωτικού κλειδιού στον αλγόριθμο ψηφιακής υπογραφής DSA (βλέπε Ενότητα 2.5) (δηλαδή, δύο υπογραφές που παράγονται για ξεχωριστά μεν μηνύματα αλλά με την ίδια τυχαία είσοδο μπορεί να οδηγούν στην άμεση ανάκτηση του ιδιωτικού κλειδιού DSA).

4.2 Ορολογία

Οι Γεννήτριες Τυχαίων Αριθμών (Random Number Generator – RNG), είναι γνωστές στη βιβλιογραφία και ως Γεννήτριες Τυχαίων Δυαδικών Ψηφίων (Random Bit Generators). Μια κατάλληλη πηγή τυχαίων δυαδικών ψηφίων μπορεί εύκολα να μετατραπεί σε μια πηγή τυχαίων αριθμών που κατανέμονται περίπου ομοιόμορφα σε ένα επιθυμητό εύρος, κάνοντας χρήση διάφορων τεχνικών [7, 8].

Οι Γεννήτριες Τυχαίων Αριθμών διακρίνονται σε δύο κύριες κατηγορίες, τις Γεννήτριες Πραγματικά Τυχαίων Αριθμών (True Random Number Generator – TRNG) και τις Γεννήτριες Ψευδοτυχαίων Αριθμών (Pseudo-Random Number Generator – PRNG). Οι TRNG συνήθως περιλαμβάνουν τη χρήση κάποιου

υλικού ειδικού σκοπού (π.χ. ηλεκτρονικά κυκλώματα και κβαντικές συσκευές) που ακολουθείται από μια κατάλληλη μετα-επεξεργασία των ακατέργαστων δεδομένων εξόδου για τη δημιουργία τυχαίων αριθμών. Όπως καθίσταται αντιληπτό, θεωρητικά όλες οι απαιτήσεις για τυχαίους αριθμούς θα μπορούσαν να καλυφθούν με τη χρήση TRNG. Άλλα, συνήθως, οι TRNG λειτουργούν με σχετικά χαμηλό ρυθμό εξόδου (σε σχέση με τις PRNG) και από την άλλη έχουν μεσαίο έως και υψηλό κόστος (σε σχέση με τις PRNG που συνήθως υλοποιούνται σε επίπεδο λογισμικού).

Μια συσκευή TRNG έχει νόημα και μπορεί να βρει εφαρμογή στην δημιουργία πολύ σημαντικών και εναίσθητων κρυπτογραφικών κλειδιών, για παράδειγμα τα κύρια κλειδιά ενός συστήματος σε ένα ασφαλές περιβάλλον, αλλά η χρήση της μπορεί να θεωρηθεί “υπερβολική” για γενική χρήση. Οι PRNG είναι κατάλληλες για περιβάλλοντα γενικής χρήσης και συνήθως βασίζονται μόνο σε λογισμικό. Σε αυτή την περίπτωση, η προσέγγιση είναι να δημιουργηθούν ντετερμινιστικά αποτελέσματα με τυχαία εμφάνιση από μια αρχική τιμή ενός σπόρου τυχαιότητας (seed). Σημειώνεται ότι σύμφωνα με το National Institute of Standards and Technology (NIST) [8], οι PRNG αναφέρονται ως Γεννήτριες Αιτιοκρατικά (ή Ντετερμινιστικά) Τυχαίων Αριθμών (Deterministic Random Number Generator – DRNG), τονίζοντας τη μη τυχαία φύση της διαδικασίας παραγωγής. Σε αυτό το κεφάλαιο, εστιάζουμε στις PRNG, καθώς οι TRNG δεν προσφέρουν γενικά την ευελιξία και το σχετικά μικρό κόστος που προσφέρουν οι PRNG που βασίζονται σε λογισμικό.

Μια PRNG συνήθως περιλαμβάνει την δυνατότητα επαναστοράς (reseeding) της γεννήτριας κάνοντας χρήση μιας νέας πηγής τυχαιότητας. Το πρόβλημα της απόκτησης μια κατάλληλης και εξασφαλισμένης υψηλής ποιότητας τυχαιότητας για τον σκοπό αυτό είναι μία από τις πιο δύσκολες πτυχές του σχεδιασμού συστημάτων που χρησιμοποιούν PRNG.

Τέλος, μερικές φορές οι PRNG, ανάλογα με τις πηγές εντροπίας που χρησιμοποιούν ως σπόρο τυχαιότητας, αναφέρονται ως εξής:

- **PRNG Αποκλεισμού (blocking):** Μπλοκάρουν την παραγωγή τυχαίων αριθμών μέχρι να βρεθεί επαρκής εντροπία από τις διάφορες πηγές τυχαιότητας.
- **PRNG Μη-αποκλεισμού (non-blocking):** Δεν μπλοκάρουν την παραγωγή τυχαίων αριθμών και κάνουν χρήση της όποιας διαθέσιμης πηγής τυχαιότητας υπάρχει.

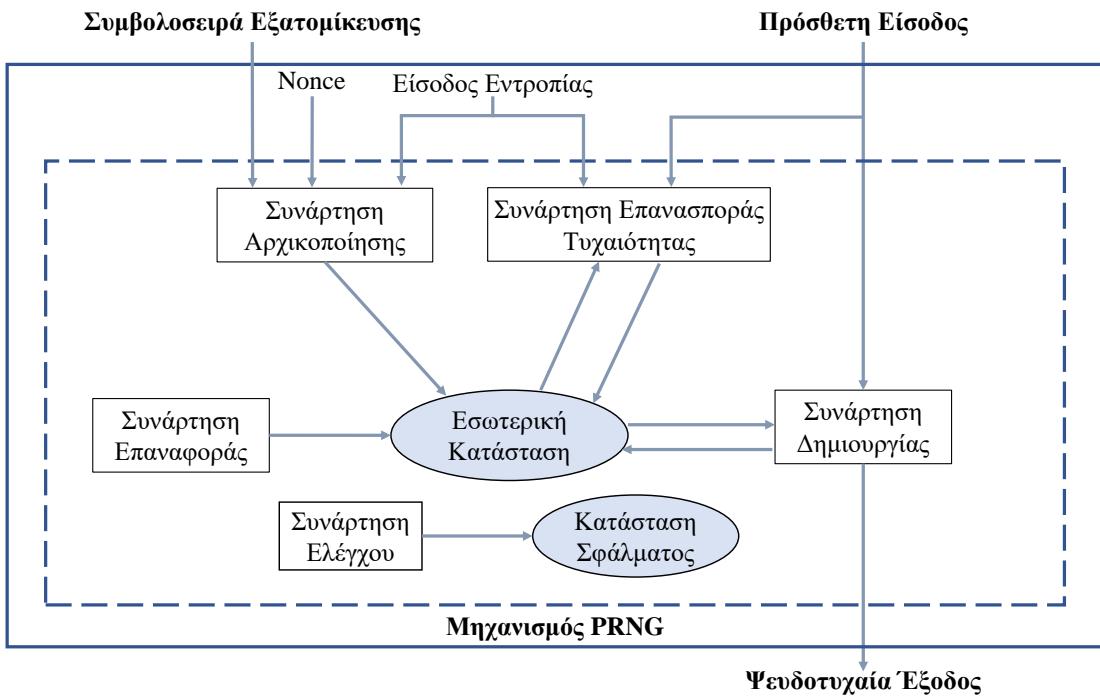
Για παράδειγμα, η PRNG στον πυρήνα του Linux παρέχει δύο διαφορετικές γεννήτριες τυχαίων αριθμών, έναν για κάθε τύπο. Επιπλέον, ο πρώτος από αυτούς τους δύο τύπους είναι αυτός που ενδείκνυνται ειδικά για κρυπτογραφικά ασφαλείς εφαρμογές.

4.3 Αρχιτεκτονική των PRNG

Μια βασική αρχιτεκτονική επιλογή που ακολουθείται από τις περισσότερες σύγχρονες PRNG είναι ο διαχωρισμός των προβλημάτων εισαγωγής εντροπίας και παραγωγής σπόρου τυχαιότητας (seed), από το πρόβλημα δημιουργίας ψευδοτυχαίων εξόδων ως συνάρτηση του σπόρου τυχαιότητας και της τρέχουσας κατάστασης της γεννήτριας. Το NIST [8] παρέχει ένα γενικό αρχιτεκτονικό μοντέλο (Σχήμα 4.1) για την περιγραφή και την ταξινόμηση των PRNG που καθιστά σαφή αυτή τη διάκριση.

Τα κύρια στοιχεία που συνθέτουν αυτό το αρχιτεκτονικό μοντέλο είναι τα εξής:

- **Είσοδος Εντροπίας:** Παρέχεται στη PRNG με σκοπό την παραγωγή του σπόρου τυχαιότητας (seed). Αυτή η είσοδος δεν είναι ομοιόμορφα τυχαία, αλλά θεωρείται ότι περιέχει αρκετή εντροπία ώστε να μπορεί να εξαχθεί ένας σπόρος τυχαιότητας κατάλληλης ποιότητας από αυτήν. Η είσοδος εντροπίας και ο σπόρος τυχαιότητας πρέπει να παραμένουν μυστικοί για να παραμείνουν ασφαλείς οι ψευδοτυχαίες έξοδοι της PRNG. Το απόρρητο αυτών των πληροφοριών παρέχει τη βάση για την ασφάλεια μιας PRNG. Η πηγή τυχαιότητας πρέπει, κατ' ελάχιστον, να παρέχει είσοδο που ικανοποιεί τις απαιτήσεις ασφάλειας του συστήματος που χρησιμοποιεί την PRNG. Η είσοδος μπορεί αρχικά να παρέχεται από



Σχήμα 4.1: Αρχιτεκτονικό μοντέλο μιας γεννήτριας ψευδοτυχαίων αριθμών.

τον χρήστη που εκτελεί την PRNG ή μπορεί να συλλέγεται από την πλατφόρμα στην οποία λειτουργεί η PRNG.

- **Άλλες Είσοδοι:** Αυτές μπορεί να βασίζονται στο χρόνο ή να λάβουν τη μορφή μιας μοναδικής τιμής (nonce). Αυτές οι πληροφορίες δεν θεωρούνται μυστικές. Συνδυάζονται με την είσοδο εντροπίας κατά τη δημιουργία του σπόρου τυχαιότητας.
- **Συμβολοσειρά Εξατομίκευσης:** Αποτελεί μια επιπλέον είσοδο στη διαδικασία παραγωγής του σπόρου τυχαιότητας, η οποία προορίζεται να προσφέρει περαιτέρω στην ποικιλομορφία των εξόδων της γεννήτριας. Για παράδειγμα, μπορούν να χρησιμοποιηθούν διαφορετικές συμβολοσειρές για τη δημιουργία κλειδιών για διαφορετικούς αλγόριθμους.
- **Εσωτερική Κατάσταση:** Αντιπροσωπεύει την εσωτερική μνήμη της PRNG, συμπεριλαμβανομένων των δεδομένων που χρησιμοποιούνται ως είσοδοι (και ενδεχομένως ακόμη και τροποποιημένων) κατά τη δημιουργία των ψευδοτυχαίων εξόδων. Σαφώς, αυτή η κατάσταση πρέπει να παραμείνει μυστική για να παραμείνουν ασφαλείς οι μελλοντικές εξόδοι του PRNG.
- **Συνάρτηση Αρχικοποίησης:** Αυτή η συνάρτηση χρησιμοποιεί την είσοδο εντροπίας, οποιαδήποτε άλλη είσοδο και τη συμβολοσειρά ξετομίκευσης, συνδυάζοντάς τες προκειμένου να δημιουργήσει έναν σπόρο τυχαιότητας από το οποίο δημιουργείται η αρχική εσωτερική κατάσταση.
- **Συνάρτηση Δημιουργίας:** Αυτή η συνάρτηση χρησιμοποιεί την τρέχουσα εσωτερική κατάσταση για να δημιουργήσει ψευδοτυχαία bit εξόδου και να ενημερώσει την κατάσταση για το επόμενο αίτημα για ψευδοτυχαία έξοδο. Η συνάρτηση θα πρέπει να διατηρεί έναν μετρητή που θα δηλώνει τον αριθμό των αιτημάτων που εξυπηρετούνται ή τα μπλοκ εξόδου που παράγονται από την πρώτη τροφοδοσία της γεννήτριας με σπόρο τυχαιότητας ή την εκ νέου τροφοδοσία της. Αυτός ο μετρητής θα επιτρέψει στη PRNG να αποκλείσει περαιτέρω αιτήματα μόλις επιτευχθεί ένα προκαθορισμένο όριο στην ποσότητα των παραγόμενων εξόδων.

- **Συνάρτηση Επανασποράς Τυχαιότητας (Reseed):** Αυτή η συνάρτηση συνδυάζει μια νέα είσοδο εντροπίας (και πιθανώς μια περαιτέρω πρόσθετη είσοδο) με την τρέχουσα εσωτερική κατάσταση για να δημιουργήσει έναν νέο σπόρο τυχαιότητας και μια νέα εσωτερική κατάσταση.
- **Συνάρτηση Επαναφοράς:** Αυτή η συνάρτηση διαγράφει την εσωτερική κατάσταση της PRNG. Η προοριζόμενη χρήση της είναι να διασφαλίσει την ασφαλή απενεργοποίηση μιας PRNG χωρίς την οποιαδήποτε διαρροή πληροφοριών που αφορά την τελευταία της κατάσταση.
- **Συνάρτηση Ελέγχου:** Αυτή η συνάρτηση προορίζεται να παρέχει έναν μηχανισμό με τον οποίο μπορεί να ελεγχθεί η σωστή δημιουργία ψευδοτυχαίων εξόδων από τη PRNG.

Τα δύο τελευταία στοιχεία συχνά δεν εμφανίζονται σε όλες τις υλοποιήσεις των PRNG. Επίσης, πολλές PRNG δεν χρησιμοποιούν άλλες εισόδους, ούτε επιτρέπουν τη χρήση συμβολοσειρών εξατομίκευσης. Ορισμένες γεννήτριες στη βιβλιογραφία [8] δε διαχωρίζουν πλήρως τις συναρτήσεις επανασποράς τυχαιότητας και δημιουργίας.

4.4 Απαιτήσεις Ασφαλείας για PRNG

Οι απαιτήσεις ασφαλείας για PRNG τυπικά καθορίζονται από τις απαιτήσεις ασφαλείας των εφαρμογών στις οποίες προορίζονται να χρησιμοποιηθούν οι ψευδοτυχαίες έξοδοι τους. Συνήθεις απαιτήσεις είναι οι εξής:

- **Δυσδιακριτότητα εξόδου (indistinguishability):** Χωρίς γνώση του αρχικού σπόρου τυχαιότητας (seed) ή/και της τρέχουσας κατάστασης, θα πρέπει να είναι δύσκολο να διακρίνουμε τις εξόδους της γεννήτριας από μια πραγματικά τυχαία ακολουθία του ίδιου τύπου, ακόμη και όταν είναι γνωστές πολλές προηγούμενες έξοδοι. Σε ορισμένες γεννήτριες, αυτή η ιδιότητα μπορεί να αποδειχθεί με βάση κάποια υπολογιστική υπόθεση πολυπλοκότητας (π.χ. οι έξοδοι της γεννήτριας Blum-Blum-Shub [9] είναι ψευδοτυχαίες, λαμβάνοντας υπόψη τη πολυπλοκότητα του προβλήματος της τετραγωνικής υπολειμματικότητας (quadratic residuosity), το οποίο σχετίζεται στενά με το πρόβλημα της παραγοντοποίησης). Για γρήγορες και αποδοτικές γεννήτριες, που βασίζονται σε συναρτήσεις σύνοψης και αλγορίθμους κρυπτογραφίας μπλοκ, αυτή η ιδιότητα βασίζεται σε μη αποδεδειγμένες αλλά εύλογες υποθέσεις ασφαλείας σχετικά με αυτά τα κρυπτογραφικά στοιχεία (π.χ. η CTR PRNG του NIST [8] παρέχει δυσδιακριτότητα εξόδου που βασίζεται σε αλγόριθμο κρυπτογραφίας μπλοκ που λειτουργεί ως μια ψευδοτυχαία συνάρτηση. Σημειώνεται ότι για έναν αλγόριθμο μπλοκ των n -bits αυτό απαιτεί να δημιουργηθούν πολύ λιγότερες από $2^{n/2}$ έξοδοι με το ίδιο κλειδί).
- **Ασφάλεια προς τα εμπρός και προς τα πίσω (forward / backward security):** Η αποκάλυψη της εσωτερικής κατάστασης της γεννήτριας δεν πρέπει να επιτρέπει σε έναν επιτιθέμενο να υπολογίζει μελλοντικές ή προηγούμενες έξόδους της γεννήτριας. Αυτή η απαίτηση συνεπάγεται ότι πρέπει να είναι δύσκολο να υπολογιστεί οποιαδήποτε μελλοντική ή προηγούμενη κατάσταση της γεννήτριας από την τρέχουσα κατάσταση.
- **Ανθεκτικότητα σε επιθέσεις επέκτασης κατάστασης:** Σε μια επίθεση επέκτασης κατάστασης [10], ένας επιτιθέμενος υποτίθεται ότι αποκτά πρόσβαση στις πληροφορίες κατάστασης της γεννήτριας και στη συνέχεια προσπαθεί να μάθει μελλοντικές έξόδους της γεννήτριας (ή να τις διακρίνει από τυχαίες). Φυσικά κάτι τέτοιο είναι δυνατόν χωρίς να υπάρχει δυνατότητα ανανέωσης της τυχαιότητας (reseed-ing), δεδομένου ότι οι μελλοντικές καταστάσεις και οι αντίστοιχες ψευδοτυχαίες έξοδοι είναι αποτέλεσμα μιας ντετερμινιστικής συνάρτησης της τρέχουσας κατάστασης. Επιπλέον, εάν χρησιμοποιηθεί η διαδικασία ανανέωσης της τυχαιότητας, αλλά δεν έχει επαρκή εντροπία στην είσοδο της, τότε ένας επιτιθέμενος μπορεί και πάλι να προσπαθήσει να υπολογίσει άλλες έξόδους, μέσω της διαδικασίας ανανέωσης της τυχαιότητας, δοκιμάζοντας όλες τις πιθανές τιμές που χρησιμοποιήθηκαν για τις άγνωστες

εισόδους εντροπίας. Είναι επιθυμητό μια PRNG να είναι ανθεκτική σε μια τέτοια επίθεση, καθώς ο στόχος της διαδικασίας ανανέωσης της τυχαιότητας είναι να αποτρέψει παραβιάσεις της κατάστασης της γεννήτριας.

Τέλος, σημειώνεται ότι καμία από αυτές τις απαιτήσεις ασφάλειας δεν αναφέρεται άμεσα στην ποιότητα των εισόδων εντροπίας, αλλά αυτό προκύπτει έμμεσα ως βασικό μέλημα για την ικανοποίηση των απαιτήσεων.

4.5 Θέματα Υλοποίησης

Εκτός από την ικανοποίηση των παραπάνω απαιτήσεων ασφαλείας, υπάρχουν πολλά ζητήματα που πρέπει να αντιμετωπιστούν κατά την υλοποίηση μιας PRNG.

4.5.1 Πηγές Εντροπίας

Τα κυριότερα από αυτά τα ζητήματα υλοποίησης αφορούν στον τρόπο προσδιορισμού κατάλληλων πηγών εντροπίας, τον τρόπο διαχείρισης και επεξεργασίας αυτών των πηγών και το πώς θα αξιολογηθεί η ποιότητα της εντροπίας που εξάγεται από αυτές τις πηγές όταν πραγματοποιείται ανανέωση της τυχαιότητας (reseeding) για την PRNG [11, 12].

4.5.2 Εκτίμηση Εντροπίας

Υπάρχει αρκετή συζήτηση στην επιστημονική κοινότητα σχετικά με το εάν μια εφαρμογή πρέπει να προσπαθήσει να εκτιμήσει πόση εντροπία είναι διαθέσιμη από τις πηγές που χρησιμοποιεί. Η ακριβής (ή τουλάχιστον συντηρητική) εκτίμηση της εντροπίας είναι σημαντική λόγω επιθέσεων επέκτασης κατάστασης: πολύ μικρή εντροπία και μια μη εξουσιοδοτημένη πρόσβαση στην κατάσταση (ή μια προεπιλεγμένη αρχική κατάσταση) της γεννήτριας μπορεί να οδηγήσει σε προβλέψιμες εξόδους. Από την άλλη, η μεγάλη διάρκεια αναμονής παρέχει κακή προστασία έναντι μη εξουσιοδοτημένης αποκάλυψης της κατάστασης της γεννήτριας, ενισχύοντας την επίθεση της ασφάλειας προς τα εμπρός (forward security). Η πλειοψηφία των PRNG κάποια μορφή εκτίμησης της εντροπίας. Ωστόσο, οι Ferguson *et al.* [7] υποστηρίζουν ότι καμία διαδικασία δεν μπορεί να εκτιμήσει με ακρίβεια την εντροπία (ή μάλλον, την ποσότητα της εντροπίας που είναι άγνωστη σε έναν εισβολέα) σε όλα τα περιβάλλοντα. Η γεννήτρια Fortuna, που προτείνουν, προσπαθεί να ξεπεράσει το πρόβλημα της εκτίμησης της εντροπίας με την κατανομή της συγκεντρωμένης εντροπίας να βασίζεται πλέον σε μια σειρά πηγών εντροπίας που αντιπροσωπεύονται από γεγονότα. Στη συνέχεια, η γεννήτρια Fortuna χρησιμοποιεί τις πηγές σε διαφορετικά διαστήματα κατά την διαδικασία ανανέωσης της τυχαιότητας. Μια ανάλυση αυτής της προσέγγισης παρέχεται στο [13].

4.5.3 Αρχικοποίηση Γεννήτριας

Μια σημαντική ειδική περίπτωση χρήσης του σπόρου τυχαιότητας είναι η ρύθμιση της αρχικής εσωτερικής κατάστασης (η οποία γίνεται μέσω της συνάρτησης αρχικοποίησης). Μια γεννήτρια δεν θα πρέπει να δημιουργεί ακολουθίες μέχρι ότου να αρχικοποιηθεί σωστά, είτε με εντροπία που παρέχεται από τον ίδιο τον χρήστη, είτε με εντροπία που συλλέγεται από το τοπικό περιβάλλον. Για παράδειγμα, στο Linux η γεννήτρια /dev/random διακόπτει την παραγωγή ακολουθιών ψευδοτυχαίων αριθμών, ακόμα και μετά την αρχικοποίηση, όποτε διαπιστωθεί ότι το μέγεθος της αιτούμενης παραγόμενης ακολουθίας ψευδοτυχαίων αριθμών υπερβαίνει το συνολικό μέγεθος της αναγκαίας ληφθείσας εντροπίας. Σε κάθε περίπτωση, η πρόσβαση σε μια γεννήτρια πριν από τη σωστή αρχικοποίησή της για πρώτη φορά έχει αναφερθεί ως μια πηγή σοβαρών προβλημάτων ασφαλείας, ιδίως κατά την δημιουργία κλειδιών [4]. Ωστόσο, είναι ασφαλές για μια γεννήτρια με το κατάλληλο σπόρο τυχαιότητας να παράγει μια μεγάλη ψευδοτυχαία ακολουθία εξόδου εάν η γεννήτρια είναι σωστά σχεδιασμένη.

4.5.4 Κρυπτογραφικά Ασφαλείς Γεννήτριες

Οι συναρτήσεις που υπάρχουν διαθέσιμες σε βιβλιοθήκες γλωσσών προγραμματισμού, όπως η `random()` στη γλώσσα προγραμματισμού C, πρέπει να αποφεύγονται σε κρυπτογραφικά συστήματα. Σε γενικές γραμμές, τέτοιες συναρτήσεις τείνουν να βασίζονται σε πολύ αδύναμες γεννήτριες, όπως απλά γραμμικές γεννήτριες αριθμών. Για τον λόγο αυτό, απαιτούνται αποκλειστικά κρυπτογραφικές υλοποιήσεις γεννητριών για συστήματα με υψηλές απαιτήσεις ασφαλείας.

Υπάρχουν πολλές γεννήτριες που χρησιμοποιούνται για τις ανάγκες των διαφόρων λειτουργικών συστημάτων, όπως η γεννήτρια των Windows και του Linux, παλαιότερες εκδόσεις των οποίων είχαν αναλυθεί στο παρελθόν. Οι αναλύσεις αυτές ανέδειξαν σοβαρές ελλείψεις και ευπάθειες, τόσο για τα Windows [14], όσο και για το Linux [15]. Επίσης, πολλές γεννήτριες παρέχονται επίσης σε κρυπτογραφικές βιβλιοθήκες. Προεξέχουσα μεταξύ αυτών είναι η γεννήτρια του OpenSSL, παλαιότερες εκδόσεις της οποίας έχουν επίσης αναλυθεί στο παρελθόν [16].

Το NIST με βάση την ειδική αναφορά 800-90A (Rev. 1) [8] καθορίζει τους μηχανισμούς για τη δημιουργία τυχαίων δυαδικών ψηφίων χρησιμοποιώντας ντετερμινιστικές μεθόδους, οι οποίοι βασίζονται σε συναρτήσεις σύνοψης και σε αλγορίθμους κρυπτογραφίας μπλοκ. Η επιλογή του κατάλληλου μηχανισμού θα πρέπει να είναι αποτέλεσμα ανάλυσης των απαιτήσεων της εφαρμογής που θα χρησιμοποιεί τους ψευδοτυχαίους αριθμούς. Πιο αναλυτικά, για κάθε έναν από αυτούς τους δύο τύπους μηχανισμών ισχύουν τα εξής:

- **Μηχανισμοί βασισμένοι σε συναρτήσεις σύνοψης:** Αποτελούν γεννήτριες αιτιοκρατικά ψευδοτυχαίων αριθμών που βασίζονται σε συναρτήσεις σύνοψης. Δύο τέτοιες γεννήτριες είναι οι:
 - HASH-DRBG
 - HMAC-DRBG

Οι οποίες έχουν σχεδιαστεί να χρησιμοποιούν οποιαδήποτε εγκεκριμένη συνάρτηση σύνοψης και μπορούν να χρησιμοποιηθούν από εφαρμογές με διάφορες απαιτήσεις ασφαλειας, με την προϋπόθεση ότι με βάση τον σπόρο τυχαιότητας επιτυγχάνεται επαρκής εντροπία.

- **Μηχανισμοί βασισμένοι σε αλγορίθμους κρυπτογραφίας μπλοκ:** Αποτελούν γεννήτριες αιτιοκρατικά ψευδοτυχαίων αριθμών που βασίζονται σε αλγορίθμους κρυπτογραφίας μπλοκ. Μια τέτοια γεννήτρια, όπως η CTR-DRBG, μπορεί να χρησιμοποιήσει τυπικά οποιονδήποτε εγκεκριμένο αλγόριθμο κρυπτογραφίας μπλοκ και να βρει χρήση σε εφαρμογές που απαιτούν διαφορετικά επίπεδα ασφαλείας. Η μόνο προϋπόθεση που υπάρχει είναι η χρήση ενός κατάλληλου αλγορίθμου κρυπτογραφίας μπλοκ σε συνδυασμό με ένα κατάλληλου μήκους κλειδιού παρέχοντας με αυτό τον τρόπο επαρκή εντροπία για τον σπόρο τυχαιότητας.

4.6 Μηχανισμοί Κρυπτογραφικά Ασφαλών PRNG

Στην ενότητα αυτή συνοψίζονται, σύμφωνα με το NIST [8], τρεις μηχανισμοί για τη δημιουργία Γεννητριών Κρυπτογραφικά Ασφαλών Ψευδοτυχαίων Αριθμών (Cryptographically Secure Pseudo-Random Number Generator – CSPRNG), δύο εκ των οποίων (HASH-DRBG και HMAC-DRBG) βασίζονται σε συναρτήσεις σύνοψης και ένας (CTR-DRBG) σε αλγορίθμους κρυπτογραφίας μπλοκ. Οι τρεις αυτοί μηχανισμοί παρουσιάζουν κάποια κοινά χαρακτηριστικά και λειτουργίες που συνοψίζονται παρακάτω:

- Βασίζονται σε μια συνάρτηση κρυπτογραφικά μονόδρομη, παρέχοντας έτσι αντίσταση υπαναχώρησης (backtrack resistance).
- Η μνήμη της εσωτερικής κατάστασης είναι μυστική και απροσπέλαστη στον χρήστη.
- Επιτρέπουν τις ακόλουθες βασικές λειτουργίες:

- *Αρχικοποίηση*, για την απόκτηση ενός σπόρου τυχαιότητας (δηλ., συνένωση της εντροπίας εισόδου, πιθανώς μιας τυχαίας εξωτερικής ή εσωτερικής μοναδικής τιμής (nonce), με μια συμβολοσειρά εξατομίκευσης) και τον καθορισμό της εσωτερικής κατάστασης σε μια τυχαία τιμή που προέρχεται από τον σπόρο τυχαιότητας.
 - *Επανασπορά τυχαιότητας*, για την απόκτηση ενός σπόρου τυχαιότητας (δηλ., συνένωση της εσωτερικής κατάστασης, της παρεχόμενης εντροπίας εισόδου με τη συμβολοσειρά εξατομίκευσης) και την ενημέρωση της εσωτερικής κατάστασης σε μια τυχαία τιμή που προέρχεται από τον σπόρο τυχαιότητας.
 - *Δημιουργία*, για την παραγωγή μιας ακολουθίας bits εξόδου με βάση την τρέχουσα κατάσταση και στη συνέχεια την ενημέρωση της επόμενης κατάστασης σε μια τυχαία τιμή που προέρχεται από την προηγούμενη κατάσταση. Στις υποενότητες (4.6.1, 4.6.2 και 4.6.3) που ακολουθούν επικεντρωνόμαστε σε αυτή μόνο την λειτουργία για λόγους συντομίας.
 - *Επαναφορά*, για την διαγραφή της εσωτερικής κατάστασης.
- Υποστηρίζουν υψηλή παρεχόμενη ασφάλεια (112, 128, 192 ή 256 bits), αλλά και χαμηλότερου επιπέδου.
 - Διαθέτουν έναν μετρητή επανασποράς τυχαιότητας και την αντίστοιχη διάρκεια ζωής των σπόρων τυχαιότητας, που σηματοδοτούν στο χρήστη ότι ο μηχανισμός χρειάζεται νέο σπόρο τυχαιότητας.
 - Ο χρήστης έχει την δυνατότητα εισαγωγής μιας συμβολοσειράς εξατομίκευσης, η οποία δεν χρειάζεται να είναι μυστική, αλλά συμβάλλει στην τυχαιοποίηση της εσωτερικής κατάστασης.

4.6.1 Μηχανισμός HASH-DRBG

Ο μηχανισμός HASH-DRBG βασίζεται σε μια κρυπτογραφική συνάντηση σύνοψης $H : \{0,1\}^* \rightarrow \{0,1\}^{outlen}$ (περισσότερα στο Κεφάλαιο 3) και διαθέτει μια εσωτερική κατάσταση $S = (V, C, cnt)$. Η εσωτερική αυτή κατάσταση S απαρτίζεται από τα εξής:

- Μια τιμή $V \in \{0,1\}^{seedlen}$ που ενημερώνεται κατά τη διάρκεια κάθε κλήσης του μηχανισμού.
- Μια σταθερά $C \in \{0,1\}^{seedlen}$ που εξαρτάται από τον σπόρο τυχαιότητας.
- Έναν μετρητή cnt που υποδεικνύει τον αριθμό των αιτημάτων για ψευδοτυχαία bits από την είσοδο μιας νέας εντροπίας που προκύπτει κατά την αρχικοποίηση ή την επανασπορά τυχαιότητας του μηχανισμού.

Οι μεταβλητές V και C αποτελούν εξαιρετικά σημαντικές παραμέτρους ασφαλείας. Εάν και η προτυποποίηση του NIST [8] δεν δηλώνει ρητά το ρόλο του C , ωστόσο, ο σκοπός του φαίνεται να είναι η αποφυγή μετάβασης του μηχανισμού HASH-DRBG σε μια σειρά επαναλαμβανόμενων καταστάσεων. Στον Πίνακα 4.1 παρουσιάζονται ενδεικτικά οι κύριες παράμετροι του μηχανισμού HASH-DRBG για την οικογένεια αλγορίθμων σύνοψης SHA-2.

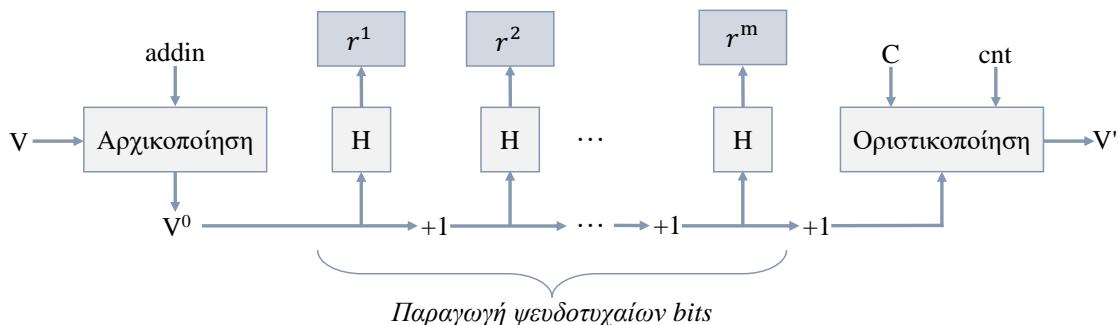
Στο Σχήμα 4.2 αποτελούνται η εξέλιξη της κατάστασης $S = (V, C, cnt)$ μετά από μια κλήση του μηχανισμού HASH-DRBG (όπως παρουσιάζεται στο [17]), ενώ στον Αλγόριθμο 4.1 παρουσιάζονται αρκετά αναλυτικά τα βήματα που ακολουθούνται για την δημιουργία ψευδοτυχαίων bits εξόδου. Η παραγωγή μιας ψευδοτυχαίας εξόδου μέσω της διαδικασίας που παρουσιάζεται στον Αλγόριθμο 4.1 πραγματοποιείται ως εξής:

- Κατά την αρχικοποίηση, εάν χρησιμοποιηθεί πρόσθετη είσοδος (*addin*) στην κλήση του μηχανισμού HASH-DRBG, παράγεται η σύνοψη της και προστίθεται στην τιμή V (γραμμές 4 - 6).
- Στη συνέχεια, παράγεται το μπλοκ εξόδου με ψευδοτυχαία bits κάνοντας χρήση μιας συνάρτησης σύνοψης H και της τιμής V που αυξάνει κατά 1 κάθε φορά (γραμμές 12 - 15).

Πίνακας 4.1: Παράμετροι του μηχανισμού HASH-DRBG για την οικογένεια συναρτήσεων σύνοψης SHA-2.

Παράμετρος	Οικογένεια Συναρτήσεων Σύνοψης SHA-2			
	SHA-224	SHA-256	SHA-384	SHA-512
Τυψηλότερη παρεχόμενη ασφάλεια	192 bits	256 bits	256 bits	256 bits
Μήκος μπλοκ εξόδου (outlen)	224 bits	256 bits	384 bits	512 bits
Ελάχιστη εντροπία για αρχικοποίηση και επανασπορά τυχαιότητας	192 bits	256 bits	256 bits	256 bits
Μήκος σπόρου τυχαιότητας (seedlen)	440 bits	440 bits	888 bits	888 bits
Μέγιστος πλήθος από bit εξόδου ανά αίτημα	2^{19}	2^{19}	2^{19}	2^{19}
Μέγιστος πλήθος αιτημάτων μεταξύ των επανασπορών τυχαιότητας	2^{48}	2^{48}	2^{48}	2^{48}

- Τέλος, κατά την οριστικοποίηση, παράγεται η σύνοψη της τιμής V με την προσάρτηση ενός κατάλληλου προθέματος (δηλ., Θχθ3) και η συμβολοσειρά που προκύπτει, μαζί με τη σταθερά C και τον μετρητή επανασποράς τυχαιότητας cpt , προστίθενται στο V για την ενημέρωση της τιμής του για την επόμενη δημιουργία ψευδοτυχαίων bits (γραμμές 18 - 19).

Σχήμα 4.2: Εξέλιξη της κατάστασης $S = (V, C, cnt)$ μετά από μια κλήση του μηχανισμού HASH-DRBG.

4.6.2 Μηχανισμός HMAC-DRBG

Ο μηχανισμός HMAC-DRBG βασίζεται σε έναν μηχανισμό HMAC (Keyed-Hash Message Authentication Code) $HMAC : \{0,1\}^{outlen} \times \{0,1\}^* \rightarrow \{0,1\}^{outlen}$. Το HMAC [18] αποτελεί έναν συγκεκριμένο τύπο Κώδικα Αυθεντικοποίησης Μηνύματος (Message Authentication Code – MAC) που περιλαμβάνει μια κρυπτογραφική συνάρτηση σύνοψης και ένα μυστικό κλειδί (περισσότερα στο Κεφάλαιο 5). Το κύριο στοιχείο του μηχανισμού HMAC-DRBG είναι η εσωτερική κατάσταση $S = (K, V, cnt)$, η οποία απαρτίζεται από τα εξής:

- Ένα κλειδί $K \in \{0,1\}^{outlen}$ που ενημερώνεται τουλάχιστον μία φορά σε κάθε κλήση του μηχανισμού.
- Μια τιμή $V \in \{0,1\}^{outlen}$ που ενημερώνεται κατά τη διάρκεια κάθε κλήσης του μηχανισμού.
- Έναν μετρητή cpt που υποδεικνύει τον αριθμό των αιτημάτων για ψευδοτυχαία bits από την είσοδο μιας νέας εντροπίας που προκύπτει κατά την αρχικοποίηση ή την επανασπορά τυχαιότητας του μηχανισμού.

Αλγόριθμος 4.1: Δημιουργία ψευδοτυχαίων bits με χρήση του μηχανισμού HASH-DRBG.

Είσοδος: $S = (V, C, cnt)$, η τρέχουσα εσωτερική κατάσταση, όπου η τιμή $V \in \{0, 1\}^{seedlen}$ ενημερώνεται κατά τη διάρκεια κάθε κλήσης, η σταθερά $C \in \{0, 1\}^{seedlen}$ εξαρτάται από τον σπόρο τυχαιότητας, και ο μετρητής cnt υποδεικνύει τον αριθμό των αιτημάτων για ψευδοτυχαία bits από την είσοδο νέας εντροπίας; β , ο αριθμός των ψευδοτυχαίων bit που θα επιστραφούν; $addin$, μια προαιρετική πρόσθετη συμβολοσειρά εισόδου;

Έξοδος: $S' = (V', C, cnt')$, η επόμενη εσωτερική κατάσταση; R , η ακολουθία ψευδοτυχαίων bits εξόδου;

1 **Function** *generate* ($S, \beta, addin$)

```

1  /* Αρχικοποίηση */
2  if  $0 \leftarrow check(S, \beta, addin)$  then
3      | return (error,  $\perp$ )
4  if  $addin \neq Null$  then
5      |  $w \leftarrow H(0x02 \parallel V \parallel addin)$ 
6      |  $V^0 \leftarrow (V + w) \bmod 2^{seedlen}$ 
7  else
8      |  $V^0 \leftarrow V$ 
9  end
10 /* Παραγωγή ψευδοτυχαίων bits */
11  $m \leftarrow \lceil \frac{\beta}{outlen} \rceil$ 
12 for  $1 \leq j \leq m$  do
13     |  $r^j \leftarrow H(V^{j-1})$ 
14     |  $V^j \leftarrow (V^{j-1} + 1) \bmod 2^{seedlen}$ 
15     |  $data \leftarrow data \parallel r^j$ 
16 end
17 /* Οριστικοποίηση */
18  $R \leftarrow leftmost(data, \beta)$ 
19  $H \leftarrow H(0x03 \parallel V^0)$ 
20  $V' \leftarrow (V^0 + H + C + cnt) \bmod 2^{seedlen}$ 
21  $cnt' \leftarrow cnt + 1$ 
22 return ( $V', C, cnt'$ ),  $R$ 
23 end

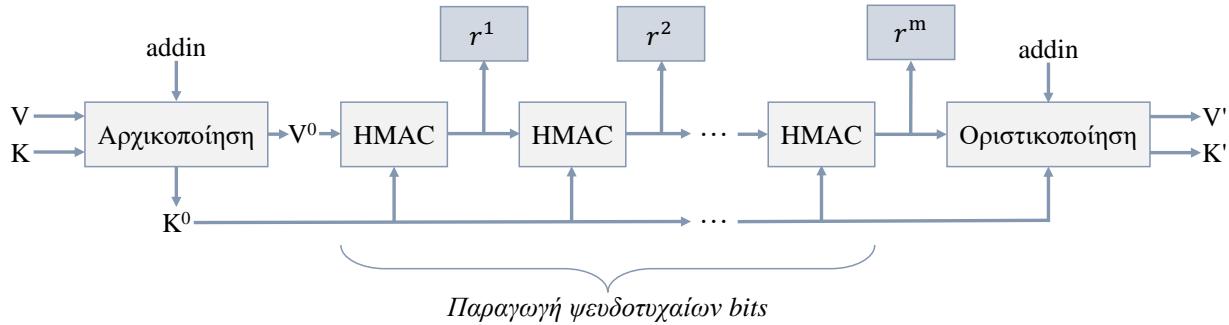
```

όπου οι μεταβλητές K και V αποτελούν εξαιρετικά σημαντικές παραμέτρους ασφαλείας του μηχανισμού HMAC-DRBG. Οι παράμετροι του μηχανισμού HMAC-DRBG είναι ανάλογες με αυτές που παρουσιάζονται στον Πίνακα 4.1.

Στο Σχήμα 4.3 αποτελούνται η εξέλιξη της κατάστασης $S = (K, V, cnt)$ μετά από μια κλήση του μηχανισμού HMAC-DRBG (όπως παρουσιάζεται στο [17]), ενώ στον Αλγόριθμο 4.2 παρουσιάζονται αρκετά αναλυτικά τα βήματα που ακολουθούνται για την δημιουργία ψευδοτυχαίων bits εξόδου. Η παραγωγή μιας ψευδοτυχαίας εξόδου μέσω της διαδικασίας που παρουσιάζεται στον Αλγόριθμο 4.2 πραγματοποιείται ως εξής:

- Κατά την αρχικοποίηση, εάν χρησιμοποιηθεί πρόσθετη είσοδος (*addin*), αυτή ενσωματώνεται στα K και V μέσω της συνάρτησης ενημέρωσης (*update*) (γραμμές 4 - 5).
- Στη συνέχεια, η ψευδοτυχαία έξοδος παράγεται μέσω του επαναληπτικού υπολογισμού $V^j \leftarrow HMAC(K^0, V^{j-1})$ και της συνένωσης των συμβολοσειρών που προκύπτουν (γραμμές 11 - 14).

- Τέλος, κατά την οριστικοποίηση, τόσο το κλειδί K όσο και η τιμή V ενημερώνονται μέσω της συνάρτησης ενημέρωσης (update) (γραμμή 17).



Σχήμα 4.3: Εξέλιξη της κατάστασης $S = (K, V, cnt)$ μετά από μια κλήση του μηχανισμού HMAC-DRBG.

4.6.3 Μηχανισμός CTR-DRBG

Ο μηχανισμός CTR-DRBG βασίζεται σε έναν αλγόριθμο κρυπτογραφίας μπλοκ $E : \{0, 1\}^{keylen} \times \{0, 1\}^{blocklen} \rightarrow \{0, 1\}^{blocklen}$ (περισσότερα στο Κεφάλαιο 1). Το κύριο στοιχείο του μηχανισμού CTR-DRBG είναι η εσωτερική κατάσταση $S = (K, V, cnt)$, η οποία απαρτίζεται από τα εξής:

- Ένα κλειδί $K \in \{0, 1\}^{keylen}$ που ενημερώνεται κάθε φορά που δημιουργείται ένας προκαθορισμένος αριθμός μπλοκ εξόδου.
- Μια τιμή $V \in \{0, 1\}^{blocklen}$ που ενημερώνεται κατά τη διάρκεια κάθε κλήσης του μηχανισμού για την δημιουργία $blocklen$ bits εξόδου.
- Έναν μετρητή cnt που υποδεικνύει τον αριθμό των αιτημάτων για ψευδοτυχαία bits από την είσοδο μιας νέας εντροπίας που προκύπτει κατά την αρχικοποίηση ή την επανασπορά τυχαιότητας του μηχανισμού.

όπου οι μεταβλητές K και V αποτελούν εξαιρετικά σημαντικές παραμέτρους ασφαλείας του μηχανισμού CTR-DRBG, οι οποίες αρχικοποιούνται σε μια ιδανικά τυχαία κατάσταση $S_0 = (K_0, V_0, cnt_0)$. Στον Πίνακα 4.2 παρουσιάζονται οι κύριες παράμετροι του μηχανισμού CTR-DRBG για τον αλγόριθμο κρυπτογραφίας μπλοκ TDEA [19] (γνωστός και ως 3DES) και την οικογένεια αλγορίθμων μπλοκ AES.

Στο Σχήμα 4.4 αποτυπώνεται η εξέλιξη της κατάστασης $S = (K, V, cnt)$ μετά από μια κλήση του μηχανισμού CTR-DRBG (όπως παρουσιάζεται στο [17]), ενώ στον Αλγόριθμο 4.3 παρουσιάζονται αρκετά αναλυτικά τα βήματα που ακολουθούνται για την δημιουργία ψευδοτυχαίων bits εξόδου. Η παραγωγή μιας ψευδοτυχαίας εξόδου μέσω της διαδικασίας που παρουσιάζεται στον Αλγόριθμο 4.3 πραγματοποιείται ως εξής:

- Κατά την αρχικοποίηση, εάν χρησιμοποιηθεί πρόσθετη είσοδος (addin) στην κλήση του μηχανισμού CTR-DRBG, αυτή ενσωματώνεται στην κατάσταση μέσω της συνάρτησης ενημέρωσης (update) (γραμμή 9). Εάν χρησιμοποιείται μια συνάρτηση παραγωγής (derivation) (περισσότερα στο Παράρτημα G του [17]), η συμβολοσειρά της πρόσθετης εισόδου διαμορφώνεται σε μια συμβολοσειρά ($keylen + blocklen$)-bits μέσω της συνάρτησης derivation πριν από αυτή τη διαδικασία (γραμμή 6). Εάν δεν χρησιμοποιείται μια συνάρτηση παραγωγής, η πρόσθετη συμβολοσειρά εισόδου ρυθμίζεται να έχει μήκος ($keylen + blocklen$)-bits.
- Στη συνέχεια, τα μπλοκ ψευδοτυχαίας εξόδου παράγονται επαναληπτικά χρησιμοποιώντας τον αλγόριθμο κρυπτογράφησης μπλοκ E σε λειτουργία μετρητή (Counter Mode – CTR) (γραμμές 16 - 19).
- Τέλος, κατά την οριστικοποίηση, τόσο το K όσο και το V ενημερώνονται μέσω μιας εφαρμογής της συνάρτησης ενημέρωσης (γραμμή 22).

Αλγόριθμος 4.2: Δημιουργία ψευδοτυχαίων bits με χρήση του μηχανισμού HMAC-DRBG.

Είσοδος: $S = (K, V, cnt)$, η τρέχουσα εσωτερική κατάσταση, όπου το κλειδί $K \in \{0, 1\}^{outlen}$ ενημερώνεται τουλάχιστον μία φορά σε κάθε κλήση, η τιμή $V \in \{0, 1\}^{outlen}$ ενημερώνεται κατά τη διάρκεια κάθε κλήσης, και ο μετρητής cnt υποδεικνύει τον αριθμό των αιτημάτων για ψευδοτυχαίων bits από την είσοδο νέας εντροπίας;
 β , ο αριθμός των ψευδοτυχαίων bit που θα επιστραφούν;
 $addin$, μια προαιρετική πρόσθετη συμβολοσειρά εισόδου;

Έξοδος: $S' = (K', V', cnt')$, η επόμενη εσωτερική κατάσταση;
 R , η ακολούθια ψευδοτυχαίων bits εξόδου;

1 **Function generate** ($S, \beta, addin$)

```

1  /* Αρχικοποίηση */
2  if  $0 \leftarrow check(S, \beta, addin)$  then
3      | return (error,  $\perp$ )
4  if  $addin \neq Null$  then
5      |  $(K^0, V^0) \leftarrow update(K, V, addin)$ 
6  else
7      |  $(K^0, V^0) \leftarrow (K, V)$ 
8  end
9  /* Παραγωγή ψευδοτυχαίων bits */
10  $m \leftarrow \lceil \frac{\beta}{outlen} \rceil$ 
11 for  $1 \leq j \leq m$  do
12     |  $V^j \leftarrow HMAC(K^0, V^{j-1})$ 
13     |  $r^j \leftarrow V^j$ 
14     |  $data \leftarrow data \parallel r^j$ 
15 end
16 /* Οριστικοποίηση */
17  $R \leftarrow leftmost(data, \beta)$ 
18  $(K', V') \leftarrow update(K^0, V^m, addin)$ 
19  $cnt' \leftarrow cnt + 1$ 
20 return  $(K', V', cnt'), R$ 
```

20 **end**21 **Function update** ($K, V, addin$)

```

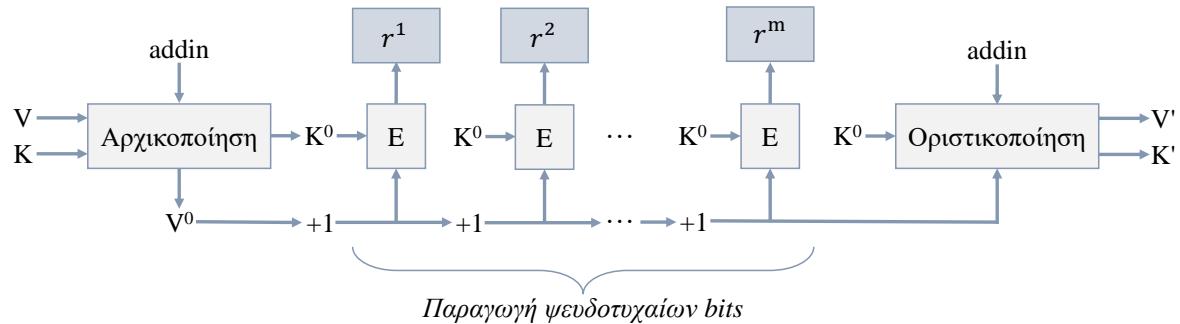
/* Ενημέρωση του K και V */
22  $K \leftarrow HMAC(K, V \parallel 0x00 \parallel addin)$ 
23  $V \leftarrow HMAC(K, V)$ 
24 if  $addin \neq Null$  then
25     |  $K \leftarrow HMAC(K, V \parallel 0x01 \parallel addin)$ 
26     |  $V \leftarrow HMAC(K, V)$ 
27 return  $(K, V)$ 
28 end
```

4.7 Άλλοι Αλγόριθμοι PRNG

Στην ενότητα αυτή παρουσιάζονται δύο αλγόριθμοι ψευδοτυχαίων αριθμών, ο Blum-Blum-Shub και ο Blum-Micali, που η ασφάλεια τους βασίζεται στο πρόβλημα της παραγοντοποίησης και του διακριτού λογάριθμου, αντίστοιχα, και επιπλέον ο αλγόριθμος του Yarrow, που ταυτόχρονα χρησιμοποιεί συνάρτηση σύνοψης και

Πίνακας 4.2: Παράμετροι του μηχανισμού CTR-DRBG (όπου $B = (2^{ctr_len} - 4) \cdot blocklen$).

Παράμετρος	3-Key TDEA	Οικογένεια Αλγορίθμων Μπλοκ AES		
		AES-128	AES-192	AES-256
Υψηλότερη παρεχόμενη ασφάλεια	112 bits	128 bits	192 bits	256 bits
Μήκος μπλοκ εισόδου/εξόδου (blocklen)	64 bits	128 bits	128 bits	128 bits
Μήκος κλειδιού (keylen)	168 bits	128 bits	192 bits	256 bits
Μήκος πεδίου μετρητή (ctr_len)		$4 < ctr_len < blocklen$		
Ελάχιστη εντροπία για αρχικοποίηση και επανασπορά τυχαιότητας	112 bits	128 bits	192 bits	256 bits
Μήκος σπόρου τυχαιότητας (seedlen)	232 bits	256 bits	320 bits	384 bits
Μέγιστος πλήθος από bit εξόδου ανά αίτημα	$\min(B, 2^{13})$	$\min(B, 2^{19})$	$\min(B, 2^{19})$	$\min(B, 2^{19})$
Μέγιστος πλήθος αιτημάτων μεταξύ των επανασπορών τυχαιότητας	2^{48}	2^{48}	2^{48}	2^{48}

Σχήμα 4.4: Εξέλιξη της κατάστασης $S = (K, V, cnt)$ μετά από μια κλήση του μηχανισμού CTR-DRBG.

κρυπτογράφηση μπλοκ για την παραγωγή ψευδοτυχαιών bits.

4.7.1 Αλγόριθμος Blum-Blum-Shub

Ο αλγόριθμος Blum-Blum-Shub [9] αποτελεί μια γεννήτρια ψευδοτυχαιών αριθμών που προτάθηκε το 1986 από τους Lenore Blum, Manuel Blum και Michael Shub. Ο αλγόριθμος αυτός υπολογίζει τον επόμενο τυχαίο αριθμό x_{n+1} με βάση την ακόλουθη εξίσωση:

$$x_{n+1} = x_n^2 \bmod M \quad (4.1)$$

όπου $M = p \cdot q$ είναι το γινόμενο δύο μεγάλων πρώτων αριθμών p και q . Σε κάθε βήμα του αλγορίθμου, η ψευδοτυχαία έξοδος προκύπτει είτε από το bit ισοτιμίας (parity) του x_{n+1} , είτε από ένα ή περισσότερα από τα λιγότερο σημαντικά bit του x_{n+1} .

Ο σπόρος τυχαιότητας x_0 πρέπει να είναι ένας ακέραιος αριθμός που είναι σχετικά πρώτος (co-prime) του M (δηλαδή το p και το q δεν είναι παράγωγοι του x_0) και όχι το 1 ή το 0. Επιπλέον, οι δύο πρώτοι, p και q , θα πρέπει και οι δύο να είναι ισοδύναμοι με το 3 (mod 4) (αυτό εγγυάται ότι κάθε τετραγωνικό υπόλοιπο (quadratic residue) έχει μια τετραγωνική ρίζα που είναι επίσης τετραγωνικό υπόλοιπο) και θα πρέπει να είναι ασφαλείς πρώτοι (safe primes) με μικρό μέγιστο κοινό διαιρέτη $\gcd((p-3)/2, (q-3)/2)$ (αυτό κάνει το μήκος κύκλου μεγάλο).

Αλγόριθμος 4.3: Δημιουργία ψευδοτυχαίων bits με χρήση του μηχανισμού CTR-DRBG.

Είσοδος: $S = (K, V, cnt)$, η τρέχουσα εσωτερική κατάσταση, όπου το κλειδί $K \in \{0,1\}^{keylen}$ ενημερώνεται κάθε φορά που δημιουργείται ένας προκαθορισμένος αριθμός μπλοκ εξόδου, η τιμή $V \in \{0,1\}^{blocklen}$ ενημερώνεται κατά τη διάρκεια κάθε κλήσης, και ο μετρητής cnt υποδεικνύει τον αριθμό των αιτημάτων για ψευδοτυχαία bits από την είσοδο νέας εντροπίας; β , ο αριθμός των ψευδοτυχαίων bit που θα επιστραφούν; $addin$, μια προαιρετική πρόσθετη συμβολοσειρά εισόδου;

Έξοδος: $S' = (K', V', cnt')$, η επόμενη εσωτερική κατάσταση; R , η ακολουθία ψευδοτυχαίων bits εξόδου;

1 **Function generate** ($S, \beta, addin$)

```

1  /* Αρχικοποίηση */
2  if  $0 \leftarrow check(S, \beta, addin)$  then
3      return (error,  $\perp$ )
4  if  $addin \neq Null$  then
5      if χρήση συνάρτησης παραγωγής then
6           $addin \leftarrow derivation(addin, (keylen + blocklen))$ 
7      else if  $len(addin) < (keylen + blocklen)$  then
8           $addin \leftarrow addin \parallel 0^{keylen+blocklen-len(addin)}$ 
9      ( $K^0, V^0$ )  $\leftarrow update(K, V, addin)$ 
10     else
11          $addin \leftarrow 0^{keylen+blocklen}$ 
12         ( $K^0, V^0$ )  $\leftarrow (K, V)$ 
13     end
14     /* Παραγωγή ψευδοτυχαίων bits */
15      $m \leftarrow \lceil \frac{\beta}{blocklen} \rceil$ 
16     for  $1 \leq j \leq m$  do
17          $V^j \leftarrow (V^{j-1} + 1) \bmod 2^{blocklen}$ 
18          $r^j \leftarrow E(K^0, V^j)$ 
19          $data \leftarrow data \parallel r^j$ 
20     end
21     /* Οριστικοποίηση */
22      $R \leftarrow leftmost(data, \beta)$ 
23     ( $K', V'$ )  $\leftarrow update(K^0, V^m, addin)$ 
24      $cnt' \leftarrow cnt + 1$ 
25     return ( $K', V', cnt'$ ),  $R$ 
26 end
27 Function update ( $K, V, addin$ )
28     /* Ενημέρωση του  $K$  και  $V$  */
29      $temp \leftarrow Null$ 
30      $m \leftarrow \lceil \frac{keylen+blocklen}{blocklen} \rceil$ 
31     for  $1 \leq j \leq m$  do
32          $V \leftarrow (V + 1) \bmod 2^{blocklen}$ 
33          $C^j \leftarrow E(K, V)$ 
```

32
 33 | temp ← temp || C^i
 34 | **end**
 35 | temp ← leftmost(temp, (keylen + blocklen))
 36 | K || V ← temp ⊕ addin
 37 | **return** (K, V)
 38 |
 39 **end**

Ένα ενδιαφέρον χαρακτηριστικό του αλγορίθμου Blum-Blum-Shub είναι ότι μπορεί να υπολογιστεί απευθείας οποιαδήποτε τιμή x_i (μέσω του θεωρήματος του Euler):

$$x_i = \left(x_0^{2^i} \bmod \lambda(M) \right) \bmod M \quad (4.2)$$

όπου λ είναι η συνάρτηση Carmichael [20], η οποία θα μπορούσε να αναχθεί σε ελάχιστο κοινό πολλαπλάσιο: $\lambda(M) = \lambda(p \cdot q) = lcm(p - 1, q - 1)$.

4.7.2 Αλγόριθμος Blum-Micali

Ο αλγόριθμος Blum-Micali [21] αποτελεί μια κρυπτογραφικά ασφαλή γεννήτρια ψευδοτυχαίων αριθμών, η ασφάλεια της οποίας βασίζεται στην δυσκολία υπολογισμού των διακριτών λογαρίθμων.

Έστω p ένας περιττός πρώτος αριθμός, το g μια πρωταρχική ρίζα (primitive root) μόντουλο p , και το x_0 ως σπόρος τυχαιότητας, τότε:

$$x_{i+1} = g^{x_i} \bmod p \quad (4.3)$$

Η έξοδος i του αλγορίθμου είναι 1 εάν $x_i \leq \frac{p-1}{2}$, διαφορετικά, η έξοδος είναι 0. Αυτό ισοδυναμεί με τη χρήση ενός bit του x_i ως τυχαίου αριθμού. Επίσης, έχει αποδειχθεί ότι μπορούν να παραχθούν $n - c - 1$ bits του x_i ως τυχαίοι αριθμοί, με την παραδοχή ότι η επίλυση του προβλήματος των διακριτών λογαρίθμων ενός n -bit ασφαλούς πρώτου αριθμού p είναι δύσκολη ακόμη και όταν ο εκθέτης είναι ένας μικρός αριθμός c -bit [22].

Για να είναι ασφαλής αυτή η γεννήτρια, ο πρώτος αριθμός p πρέπει να είναι αρκετά μεγάλος, ώστε να μην είναι εφικτός ο υπολογισμός των διακριτών λογαρίθμων μόντουλο p . Πιο συγκεκριμένα, οποιαδήποτε μέθοδος θα μπορούσε να προβλέψει τους αριθμούς που παράγονται, αυτό θα οδηγούσε σε έναν αλγόριθμο που θα έλυνε το πρόβλημα των διακριτών λογαρίθμων για αυτόν τον πρώτο αριθμό [21].

4.7.3 Αλγόριθμος Yarrow

Ο αλγόριθμος Yarrow [23] αποτελεί μια οικογένεια Κρυπτογραφικών Γεννητριών Ψευδοτυχαίων Αριθμών (Cryptographic Pseudo-Random Number Generator – CPRNG) που επινοήθηκε από τους John Kelsey, Bruce Schneier και Niels Ferguson και δημοσιεύτηκε το 1999. Μια βελτιωμένη εκδοχή του, αποτελεί ο αλγόριθμος Fortuna, που προτάθηκε από τους Ferguson και Schneier [24].

Ο αλγόριθμος Yarrow χρησιμοποιήθηκε στο λειτουργικό FreeBSD, αλλά πλέον έχει αντικατασταθεί από το Fortuna [25]. Ο Yarrow ενσωματώθηκε επίσης στα λειτουργικά iOS και macOS, αλλά η Apple έχει πλέον μεταβεί στο αλγόριθμο Fortuna από το 1ο τρίμηνο του 2020 [26].

Οι βασικές αρχές σχεδιασμού του αλγορίθμου Yarrow είναι: αντοχή σε επιθέσεις, εύκολη χρήση από προγραμματιστές χωρίς υπόβαθρο κρυπτογραφίας και δυνατότητα επαναχρησιμοποίησης υπαρχόντων δομικών στοιχείων. Ο Yarrow στοχεύει στο να παρέχει εύκολη ενσωμάτωση, επιτρέποντας στους σχεδιαστές συστημάτων να έχουν ελάχιστη γνώση της λειτουργικότητας γεννητριών ψευδοτυχαίων αριθμών.

Ο σχεδιασμός του αλγορίθμου Yarrow αποτελείται από τα ακόλουθα τέσσερα κύρια στοιχεία:

- Συσσωρευτής Εντροπίας:** Ο Yarrow συσσωρεύει την εντροπία με δύο προσεγγίσεις: (1) την γρήγορη συσσώρευση, η οποία παρέχει συχνές επανασπορές τυχαιότητας του κλειδιού για να κρατήσει τη διάρκεια των κλειδιών όσο το δυνατόν συντομότερη, (2) την αργή συσσώρευση, η οποία παρέχει αραιές αλλά συντηρητικές επανασπορές τυχαιότητας του κλειδιού. Αυτό διασφαλίζει ότι η επανασπορά τυχαιότητας είναι ασφαλής ακόμη και όταν οι εκτιμήσεις για την εντροπία είναι πολύ αισιόδοξες.
- Μηχανισμός Επανασποράς Τυχαιότητας:** Αυτό το στοιχείο συνδέει τον συσσωρευτή εντροπίας με τον μηχανισμό δημιουργίας. Η επανασπορά τυχαιότητας με βάση την γρήγορη συσσώρευση χρησιμοποιεί το τρέχον κλειδί και την σύνοψη (hash) όλων των εισόδων στην γρήγορη συσσώρευση από την έναρξη της λειτουργίας της για τη δημιουργία ενός νέου κλειδιού. Η επανασπορά τυχαιότητας με βάση την αργή συσσώρευση συμπεριφέρεται παρόμοια, εκτός από το ότι χρησιμοποιεί επίσης την σύνοψη όλων των εισόδων στην αργή συσσώρευση για να δημιουργήσει ένα νέο κλειδί. Και οι δύο επανασπορές τυχαιότητας μηδενίζουν την εκτίμηση της γρήγορης συσσώρευσης, αλλά η τελευταία μηδενίζει επίσης την εκτίμηση της αργής συσσώρευσης. Ο μηχανισμός επανασποράς τυχαιότητας ενημερώνει συνεχώς το κλειδί, έτσι ώστε ακόμα και αν το κλειδί της συσσώρευσης πληροφοριών γίνει γνωστό στον εισβολέα πριν από την επανασπορά τυχαιότητας, θα είναι άγνωστο στον εισβολέα μετά την επανασπορά τυχαιότητας. Στην περίπτωση του Yarrow-160, ο μηχανισμός επανασποράς τυχαιότητας χρησιμοποιεί την συνάρτηση σύνοψης SHA-1 και την κρυπτογράφηση μπλοκ 3-key TDEA [19] (τα λεπτομερή βήματα της διαδικασίας είναι διαθέσιμα εδώ [23]).
- Έλεγχος Επανασποράς Τυχαιότητας:** Το στοιχείο αυτό εκμεταλλεύεται την συχνή επανασπορά τυχαιότητας, η οποία είναι επιθυμητή αλλά μπορεί να επιτρέψει επαναληπτικές εικασίας (guessing attacks), και την όχι συχνή επανασπορά τυχαιότητας, η οποία διακυβεύει περισσότερες πληροφορίες σε έναν εισβολέα που κατέχει το κλειδί. Ο Yarrow χρησιμοποιεί την γρήγορη συσσώρευση για επανασπορά τυχαιότητας κάθε φορά που η πηγή υπερβαίνει ορισμένες οριακές τιμές και χρησιμοποιεί την αργή συσσώρευση για επανασπορά τυχαιότητας κάθε φορά που τουλάχιστον δύο από τις πηγές του περνούν κάποια άλλη οριακή τιμή.
- Μηχανισμός Δημιουργίας:** Στην περίπτωση του Yarrow-160, γίνεται χρήσης της κρυπτογράφησης μπλοκ 3-key TDEA [19] σε λειτουργία μετρητή (counter mode) για τη δημιουργία των μπλοκ εξόδου. Το C είναι μια τιμή μετρητή n -bits και το K είναι το κλειδί. Για την δημιουργία του επόμενου μπλοκ εξόδου, ο Yarrow πραγματοποιεί τις ακόλουθες λειτουργίες:

$$\text{Τιμή μετρητή: } C \leftarrow (C + 1) \bmod 2^n$$

$$\text{Μπλοκ εξόδου: } R \leftarrow E_K(C)$$

$$\text{Νέο κλειδί: } K \leftarrow \text{Τα επόμενα } k\text{-bits εξόδου της PRNG}$$

Ο Yarrow κρατά την καταμέτρηση του μπλοκ εξόδου, γιατί μόλις το κλειδί παραβιαστεί, η διαρροή του μπλοκ εξόδου πριν από το παραβιασμένο μπλοκ μπορεί να σταματήσει αμέσως. Μόλις επιτευχθεί κάποια παράμετρος ασφαλείας συστήματος P_g , ο αλγόριθμος θα δημιουργήσει k -bits εξόδου της PRNG και θα τα χρησιμοποιήσει ως το νέο κλειδί. Στο Yarrow-160, η παράμετρος ασφαλείας του συστήματος έχει οριστεί να είναι το 10, που σημαίνει ότι $P_g = 10$. Η παράμετρος αυτή έχει ρυθμιστεί σκοπίμα να είναι χαμηλή για να ελαχιστοποιηθεί ο αριθμός των μπλοκ εξόδου που μπορούν να ανακληθούν.

4.8 Ασκήσεις-Εργασίες

Ασκήσεις

- 4.8.1 Θεωρήστε ότι έχουμε τον αλγόριθμο Blum-Blum-Shub και ότι έχουμε επιλέξει ως πρώτους αριθμούς το $p = 11$ και το $q = 23$, για τους οποίους ισχύει ότι είναι ισοδύναμοι με το 3 ($\bmod 4$) και έχουν μέγιστο**

κοινό διαιρέτη το $gcd((p - 3)/2, (q - 3)/2) = 2$. Επιπλέον, θεωρήστε ότι έχουμε ως σπόρο τυχαιότητας το $s = x_{-1} = 3$. Ποια είναι τα πρώτα 6 bits που παράγονται με βάση:

- (1) το bit ισοτιμίας του x_{n+1} ;
- (2) το λιγότερο σημαντικό bit του x_{n+1} ;

4.8.2 Θεωρήστε ότι έχουμε τον αλγόριθμο Blum-Micali και ότι έχουμε επιλέξει ως περιττό πρώτο αριθμό το $p = 17$, το $g = 6$ ως πρωταρχική ρίζα μόντουλο p , και ως σπόρο τυχαιότητας το $x_0 = 9$. Ποια είναι τα πρώτα 4 bits που παράγονται με βάση αυτόν τον αλγόριθμο;

Εργασίες

4.8.1 Σε αυτήν την εργασία θα δοκιμάσετε διάφορους μηχανισμούς κρυπτογραφικά ασφαλών PRNGs που προτείνονται από το NIST [8], κάνοντας χρήση της γλώσσας προγραμματισμού Java και το περιβάλλον ανάπτυξης Eclipse IDE for Java Developers. Πιο αναλυτικά, θα δοκιμάσετε τους μηχανισμούς HASH-DRBG, HMAC-DRBG και CRT-DRBG. Κάνοντας χρήση του Eclipse Project “[crypto_chap04_DRBG](#)”, οι δοκιμές που θα πρέπει να γίνουν είναι οι εξής:



- (1) Για να δοκιμάσετε τους τρεις αυτούς μηχανισμούς εκτελέστε το αρχείο TestDRBG.java. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να αλλάξετε το πλήθος των bytes εξόδου που παράγονται από 32 σε 64 bytes.
- (2) Στο ίδιο αρχείο TestDRBG.java, δοκιμάστε να αυξήσετε την παρεχόμενη ασφάλεια στον μηχανισμό HASH-DRBG από 256 bits σε 512 bits. Τι θα συμβεί και γιατί;
- (3) Στην συνέχεια δοκιμάστε να μειώσετε την εντροπία που απαιτείται στον μηχανισμό CTR-DRBG από 256 bits σε 128 bits. Επιπλέον, δοκιμάστε να την αυξήσετε στα 1024 bits. Πως μπορεί δικαιολογηθεί αυτή η συμπεριφορά;
- (4) Τέλος, υλοποίήστε μια συνάρτηση με όνομα buildCTR_3DES_DRBG η οποία θα κάνει τις απαραίτητες ρυθμίσεις παρεχόμενης ασφάλειας, απαιτούμενης εντροπίας και χρήσης συμβολοσειράς εξατομίκευσης για τον μηχανισμό CTR-DRBG, κάνοντας χρήση της κρυπτογράφησης μπλοκ Triple DES. Επιπλέον, δοκιμάστε να την καλέσετε μέσα από την συνάρτηση main παράγοντας δυο τυχαίες ακολουθίες από bytes.

4.8.2 Σε αυτήν την εργασία θα δοκιμάσετε δύο εναλλακτικούς αλγορίθμους ψευδοτυχαίων αριθμών, κάνοντας χρήση της γλώσσας προγραμματισμού Java και το περιβάλλον ανάπτυξης Eclipse IDE for Java Developers. Πιο αναλυτικά, θα δοκιμάσετε τους αλγορίθμους Blum-Blum-Shub και Yarrow. Κάνοντας χρήση του Eclipse Project “[crypto_chap04_BBS_Yarrow](#)”, οι δοκιμές που θα πρέπει να γίνουν είναι οι εξής:



- (1) Για να δοκιμάσετε τον αλγόριθμο Blum-Blum-Shub πρέπει να εκτελέσετε το αρχείο TestBBS.java. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να αλλάξετε το μέγεθος του μόντουλο που χρησιμοποιείται από 512 σε 1024. Παρατηρείτε κάποια διαφορά στην λειτουργία του;
- (2) Για να δοκιμάσετε τον αλγόριθμο Yarrow πρέπει να εκτελέσετε το αρχείο TestYarrow.java. Στην συγκεκριμένη υλοποίηση ο αλγόριθμος κρυπτογραφίας μπλοκ που χρησιμοποιείται είναι αυτούς του Rijndael (δηλ., ο AES-128) σε αντίθεση με την αρχική υλοποίηση του Yarrow. Επιπλέον, τροποποιήστε τον κώδικα που σας δίνεται, προσπαθώντας να παράγετε Integers αντί για BigIntegers.

Βιβλιογραφία

- [1] Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 2018.
- [2] Ian Goldberg and David Wagner. “Randomness and the Netscape Browser”. In: *Dr. Dobb's Journal* 21 (Jan. 1996), p. 4. ISSN: 1044-789X.
- [3] Luciano Bello. *OpenSSL – Predictable Random Number Generator*. Debian Security Advisory, DSA-1571-1. <https://www.debian.org/security/2008/dsa-1571>. 2008.
- [4] Nadia Heninger, Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. “Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices”. In: *21st USENIX Security Symposium (USENIX Security 12)*. Bellevue, WA: USENIX Association, Aug. 2012, pp. 205–220. ISBN: 978-931971-95-9.
- [5] Arjen K. Lenstra et al. “Public Keys”. In: *Advances in Cryptology -- CRYPTO 2012*. Ed. by Reihaneh Safavi-Naini and Ran Canetti. Vol. 7417. LNCS. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 626–642. ISBN: 978-3-642-32009-5. doi: 10.1007/978-3-642-32009-5_37.
- [6] Thomas Ristenpart and Scott Yilek. “When Good Randomness Goes Bad: Virtual Machine Reset Vulnerabilities and Hedging Deployed Cryptography”. In: *Network and Distributed System Security (NDSS '10)*. The Internet Society, 2010.
- [7] Niels Ferguson, Bruce Schneier, and Tadayoshi Kohno. *Cryptography Engineering: Design Principles and Practical Applications*. Indianapolis, IN: Wiley Publishing, Inc., 2010. ISBN: 978-0-470-47424-2.
- [8] Elaine B. Barker and John M. Kelsey. “Recommendation for Random Number Generation Using Deterministic Random Bit Generators”. In: *NIST Special Publication 800-90A Rev 1* (2015), pp. 1–101. doi: 10.6028/NIST.SP.800-90Ar1.
- [9] Lenore Blum, Manuel Blum, and Michael Shub. “A Simple Unpredictable Pseudo-Random Number Generator”. In: *SIAM Journal on Computing* 15.2 (1986), pp. 364–383. doi: 10.1137/0215025.
- [10] John Kelsey, Bruce Schneier, David Wagner, and Chris Hall. “Cryptanalytic Attacks on Pseudo-random Number Generators”. In: *Fast Software Encryption*. Ed. by Serge Vaudenay. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 168–188. ISBN: 978-3-540-69710-7. doi: 10.1007/3-540-69710-1_12.
- [11] Donald E. Eastlake 3rd, Steve Crocker, and Jeffrey I. Schiller. *Randomness Requirements for Security*. RFC 4086. <https://www.rfc-editor.org/rfc/rfc4086.txt>. RFC Editor, June 2005. doi: 10.17487/RFC4086.
- [12] Peter Gutmann. “Software Generation of Practically Strong Random Numbers”. In: *7th USENIX Security Symposium (USENIX Security 98)*. San Antonio, TX: USENIX Association, Jan. 1998.
- [13] Yevgeniy Dodis, Adi Shamir, Noah Stephens-Davidowitz, and Daniel Wichs. “How to Eat Your Entropy and Have it Too: Optimal Recovery Strategies for Compromised RNGs”. en. In: *Algorithmica* 79.4 (Dec. 2017), pp. 1196–1232. ISSN: 0178-4617, 1432-0541. doi: 10.1007/s00453-016-0239-3.
- [14] Leo Dorrendorf, Zvi Guterman, and Benny Pinkas. “Cryptanalysis of the Random Number Generator of the Windows Operating System”. en. In: *ACM Transactions on Information and System Security* 13.1 (Oct. 2009), pp. 1–32. ISSN: 1094-9224. doi: 10.1145/1609956.1609966.

- [15] Zvi Guterman, Benny Pinkas, and Tzachi Reinman. “Analysis of the Linux Random Number Generator”. In: *Symposium on Security and Privacy (S&P '06)*. Berkeley/Oakland, CA: IEEE, 2006, p. 15. ISBN: 978-0-7695-2574-7. doi: 10.1109/SP.2006.5.
- [16] Falko Strenzke. “An Analysis of OpenSSL’s Random Number Generator”. en. In: *Advances in Cryptology -- EUROCRYPT 2016*. Ed. by Marc Fischlin and Jean-Sébastien Coron. Vol. 9665. LNCS. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 644–669. ISBN: 978-3-662-49889-7. doi: 10.1007/978-3-662-49890-3_25.
- [17] Joanne Woodage and Dan Shumow. *An Analysis of the NIST SP 800-90A Standard*. Cryptology ePrint Archive, Paper 2018/349. <https://ia.cr/2018/349>. 2018.
- [18] Hugo Krawczyk, Mihir Bellare, and Ran Canetti. *HMAC: Keyed-Hashing for Message Authentication*. RFC 2104. <https://www.rfc-editor.org/rfc/rfc2104.txt>. RFC Editor, Feb. 1997. doi: 10.17487/RFC2104.
- [19] Elaine B. Barker and Nicky W. Mouha. “Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher”. In: *NIST Special Publication 800-67 Rev 2* (2017), pp. 1–25. doi: 10.6028/NIST.SP.800-67r2.
- [20] Robert D. Carmichael. “Note on a New Number Theory Function”. In: *Bulletin of the American Mathematical Society* 16.5 (1910), pp. 232–238.
- [21] Manuel Blum and Silvio Micali. “How to Generate Cryptographically Strong Sequences of Pseudorandom Bits”. In: *SIAM Journal on Computing* 13.4 (1984), pp. 850–864. doi: 10.1137/0213053.
- [22] Rosario Gennaro. “An Improved Pseudo-Random Generator based on the Discrete Logarithm Problem”. In: *Journal of Cryptology* 18.2 (2005), pp. 91–110. doi: 10.1007/s00145-004-0215-y.
- [23] John Kelsey, Bruce Schneier, and Niels Ferguson. “Yarrow-160: Notes on the Design and Analysis of the Yarrow Cryptographic Pseudorandom Number Generator”. In: *Selected Areas in Cryptography*. Ed. by Howard Heys and Carlisle Adams. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 13–33. ISBN: 978-3-540-46513-3. doi: 10.1007/3-540-46513-8_2.
- [24] Niels Ferguson and Bruce Schneier. *Practical Cryptography*. Vol. 141. Wiley New York, 2003. ISBN: 0-471-22894-X.
- [25] The FreeBSD Project. *Huge Cleanup of Random(4) Code*. FreeBSD SVN Repository, Revision 284959. <https://svnweb.freebsd.org/base?view=revision&revision=284959>. 2015.
- [26] Apple Inc. *Apple Platform Security*. Apple Guide for Platform Security. https://help.apple.com/pdf/security/en_GB/apple-platform-security-guide-b.pdf. 2022.

ΚΕΦΑΛΑΙΟ 5

ΑΚΕΡΑΙΟΤΗΤΑ ΚΑΙ ΑΥΘΕΝΤΙΚΟΠΟΙΗΣΗ ΔΕΔΟΜΕΝΩΝ

Περίληψη

Οι μηχανισμοί ακεραιότητας και αυθεντικοποίησης δεδομένων διασφαλίζουν την προστασία από μη εξουσιοδοτημένη τροποποίηση των δεδομένων και παρέχουν εγγυήσεις αναφορικά με την πηγή τους [1]. Στο κεφάλαιο αυτό παρουσιάζονται και αναλύονται οι μηχανισμοί που καλύπτουν αυτές τις απαιτήσεις και βασίζονται τόσο σε συμμετρική κρυπτογραφία όσο και σε κρυπτογραφία δημοσίου κλειδιού, όπως είναι ο Κώδικας Αυθεντικοποίησης Μηνύματος (Message Authentication Code – MAC) (Ενότητα 5.2) και η Ψηφιακή Υπογραφή (Digital Signature) (Ενότητα 5.4). Επιπλέον, στην Ενότητα 5.3 αναλύονται οι μηχανισμοί που βασίζονται σε Αυθεντικοποιημένη Κρυπτογράφηση (Authenticated Encryption) όπως είναι ο συνδυασμός μεθόδων κρυπτογράφησης και ψηφιακών υπογραφών ή κρυπτογράφησης και MAC, καθώς επίσης και οι μηχανισμοί Offset Codebook (OCB) και Galois/Counter Mode (GCM).

Προαπαιτούμενη γνώση: Κατανόηση των βασικών εννοιών της Κρυπτογραφίας που παρατίθενται στα εισαγωγικά κεφάλαια (Κεφάλαιο 1 έως 3) αυτού του βιβλίου.

5.1 Εισαγωγή

Οι μηχανισμοί ακεραιότητας και αυθεντικοποίησης μηνυμάτων παρέχουν τα μέσα για την ανίχνευση μη εξουσιοδοτημένης δημιουργίας και τροποποίησης μηνύματος. Αυτοί οι μηχανισμοί δεν παρέχουν μόνο προστασία σε πραγματικό χρόνο, αλλά δίνουν επιπλέον στις εξουσιοδοτημένες οντότητες τη δυνατότητα ελέγχου της ακεραιότητας των δεδομένων και της επαλήθευσης της πηγής του ληφθέντος μηνύματος, και σε μεταγενέστερο χρόνο. Έτσι, για παράδειγμα, ένας παραλήπτης ενός μηνύματος ήλεκτρονικού ταχυδρομείου μπορεί να ελέγξει την ακεραιότητα του μηνύματος, ακόμη και μετά τη λήψη αυτού όταν παραστεί η ανάγκη, και όχι μόνο κατά τη λήψη αυτού.

Τύποι μηχανισμών που παρέχουν έλεγχο ακεραιότητας και αυθεντικοποίηση δεδομένων είναι:

- Κώδικας αυθεντικοποίησης μηνύματος (Message Authentication Code – MAC),

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

- Αυθεντικοποιημένη κρυπτογράφηση (Authenticated Encryption), και
- Ψηφιακή υπογραφή (Digital signature).

Καθένας από αυτούς τους μηχανισμούς έχει τα δικά του πλεονεκτήματα και μειονεκτήματα και μπορεί επίσης να έχει επιπλέον ιδιότητες, όπως αναλύονται στις ακόλουθες ενότητες.

5.2 Κώδικας Αυθεντικοποίησης Μηνύματος

Οι Κώδικες Αυθεντικοποίησης Μηνύματος (Message Authentication Code – MAC) αποτελούν κρυπτογραφικούς μηχανισμούς συμμετρικού κλειδιού που παρέχουν εγγύησης αναφορικά με την πηγή και την ακεραιότητα ενός μηνύματος. Μπορούν να θεωρηθούν ως μονόδρομες συναρτήσεις σύνοψης με κλειδί καθώς φέρουν ίδιες ιδιότητες με αυτές, αλλά επιπλέον χρησιμοποιούν συμμετρικά κρυπτογραφικά κλειδιά για τη δημιουργία και την επαλήθευσή τους. Παρέχουν ισχυρότερες διαβεβαιώσεις για την ακεραιότητα των δεδομένων από ένα άθροισμα ελέγχου (checksum) ή έναν κωδικό ανίχνευσης σφαλμάτων (error detection code), καθώς η επαλήθευση αυτών έχει σχεδιαστεί για να ανιχνεύει μόνο τυχαίες τροποποιήσεις των δεδομένων, ενώ τα MACs έχουν σχεδιαστεί για να ανιχνεύουν, πέραν των τυχαίων, και σκόπιμες, μη εξουσιοδοτημένες τροποποιήσεις των δεδομένων.

Ένας κώδικας αυθεντικοποίησης μηνύματος τυπικά χρησιμοποιείται για να προστατέψει τα μηνύματα που ανταλλάσσονται δύο οντότητες. Η δημιουργία και η επαλήθευση του απαιτεί τη χρήση ενός κοινού μυστικού κλειδιού το οποίο μοιράζεται μεταξύ του αποστολέα και του παραλήπτη. Η διαδικασία δημιουργίας MAC περιλαμβάνει το συνδυασμό του μηνύματος με το μυστικό κλειδί χρησιμοποιώντας έναν συγκεκριμένο αλγόριθμο. Το αποτέλεσμα είναι ο κώδικας αυθεντικοποίησης μηνύματος, γνωστός και ως ετικέτα (tag), για το συγκεκριμένο μήνυμα, ο οποίος τυπικά συνοδεύει το μήνυμα που προστατεύει. Στη συνέχεια ο παραλήπτης, καθώς και οποιαδήποτε εξουσιοδοτημένη οντότητα η οποία έχει στη διάθεση της το συμμετρικό κλειδί, μπορεί να εφαρμόσει τη διαδικασία επαλήθευσης στα ληφθέντα δεδομένα και στο ληφθέν MAC. Η διαδικασία επαλήθευσης απαιτεί τον επανυπολογισμό του MAC από τον παραλήπτη και τη σύγκριση της υπολογισμένης τιμής με τη ληφθείσα. Η επιτυχής επαλήθευση παρέχει αποδείξεις για την πηγή των δεδομένων (μόνο ο κάτοχος του συμμετρικού κλειδιού θα μπορούσε να δημιουργήσει το συγκεκριμένο MAC), καθώς και για την ακεραιότητα αυτών.

Δεδομένου ότι η δημιουργία και επαλήθευση MAC βασίζεται στη χρήση συμμετρικού κρυπτογραφικού κλειδιού, η επαλήθευσή του δεν παρέχει αδιαμφισβήτητη αναγνώριση της ταυτότητας της πηγής ενός μηνύματος, καθώς οι οντότητες που συμμετέχουν σε μια επικοινωνία και μοιράζονται ένα κοινό κλειδί για δημιουργία MAC, δημιουργούν για το ίδιο μήνυμα ίδιες τιμές MAC. Ως αποτέλεσμα, ένα MAC δε συνδέεται με μια μόνο οντότητα και, επομένως, δεν παρέχει μη αποποίηση πηγής, μια ιδιότητα που είναι πολύ σημαντική σε κάποιες ανταλλαγές δεδομένων. Σημειώνεται πως η μη αποποίηση πηγής εξασφαλίζεται με τις ψηφιακές υπογραφές. Ωστόσο, η χρήση των ψηφιακών υπογραφών δεν αποτελεί πάντα τη βέλτιστη λύση, κυρίως λόγω της καθυστέρησης που εισάγει η διαδικασία δημιουργίας και επαλήθευσης υπογραφής. Εκεί όπου η ταχύτητα στην ανταλλαγή δεδομένων αποτελεί προτεραιότητα, ενώ δεν απαιτείται η μη αποποίηση πηγής, η χρήση των MAC αποτελεί μονόδρομο, όπως συμβαίνει στα περισσότερα πρωτόκολλα επικοινωνίας.

Τα MAC προσφέρουν τα ακόλουθα πλεονεκτήματα έναντι των ψηφιακών υπογραφών:

- **Αποδοτικότητα:** Τα MAC είναι συνήθως ταχύτερα και απαιτούν λιγότερους πόρους από τις ψηφιακές υπογραφές καθώς έχουν λιγότερες απαιτήσεις σε υπολογιστική ισχύ.
- **Απλότητα:** Τα MAC είναι ευκολότερα στην υλοποίηση και εφαρμογή από τις ψηφιακές υπογραφές και απαιτούν λιγότερο περίπλοκη υποδομή για την επαλήθευση τους.
- **Ασφάλεια:** Τα MAC είναι λιγότερο επιρρεπή σε ορισμένους τύπους επιθέσεων, όπως επιθέσεις σύ-

γκρουσης¹ (collision attacks), επειδή χρησιμοποιούν κρυπτογράφηση συμμετρικού κλειδιού.

Τα MAC ωστόσο έχουν και ορισμένα μειονεκτήματα σε σχέση με τις ψηφιακές υπογραφές:

- Διαχείριση κλειδιών:** Καθώς τα MAC χρησιμοποιούν συμμετρική κρυπτογράφηση, ο αποστολέας και ο παραλήπτης πρέπει να μοιράζονται το ίδιο κλειδί, κάτι που μπορεί να είναι δύσκολο διαχειριστικά σε μεγάλα συστήματα με πολλούς χρήστες ή με χρήστες οι οποίοι δεν κάνουν χρήση μιας κοινής υποδομής.
- Μη αποποίηση πηγής:** Τα MAC, όπως ήδη αναφέρθηκε, δεν παρέχουν μη αποποίηση, που σημαίνει ότι ο αποστολέας του μηνύματος μπορεί μεταγενέστερα να αρνηθεί την αποστολή του μηνύματος.
- Περιορισμοί στη χρήση:** Τα MAC δεν είναι κατάλληλα για περιπτώσεις όπου πολλά μέρη πρέπει να επαληθεύσουν την αυθεντικοποίηση ενός μηνύματος.

Τα σχήματα MAC εμπίπτουν σε μία από τις ακόλουθες δύο κατηγορίες: αυτά που βασίζονται σε συμμετρική κρυπτογραφία και αυτά που βασίζονται σε συναρτήσεις σύνοψης. Ο Πίνακας 5.1 παρουσιάζει σχήματα τα οποία συνιστώνται για υπάρχοντα συστήματα (παλαιού τύπου) καθώς και για μελλοντική χρήση.

Πίνακας 5.1: Σχήματα MAC.

Σχήμα	Κατηγοριοποίηση Παλαιού Τύπου	Κατηγοριοποίηση Μελλοντική Χρήση	Δομικό Στοιχείο
CMAC	✓	✓	Οποιοσδήποτε κρυπτ. αλγόριθμος μπλοκ
EMAC	✓	✓	Οποιοσδήποτε κρυπτ. αλγόριθμος μπλοκ
AMAC	✓	✓	Οποιοσδήποτε κρυπτ. αλγόριθμος μπλοκ
HMAC	✓	✓	Οποιαδήποτε συνάρτηση σύνοψης
UMAC	✓	✓	Εσωτερική καθολική συνάρτηση σύνοψης
GMAC	✓	✗	Λειτουργίες πεπερασμένου πεδίου
Poly1305	✓	✗	Λειτουργίες πεπερασμένου πεδίου

5.2.1 MAC Βασισμένα σε Κρυπτογραφικούς Αλγορίθμους Μπλοκ

Τα σχήματα MAC που βασίζονται σε κρυπτογραφικούς αλγορίθμους μπλοκ έχουν πολλές ομοιότητες καθώς τα περισσότερα από αυτά υιοθετούν τη χρήση ενός συγκεκριμένου τρόπου λειτουργίας, του CBC-MAC. Οι διαφορές τους σχετίζονται κυρίως με τη μέθοδο πλήρωσης (padding) που χρησιμοποιείται για την κρυπτογράφηση του μηνύματος, τον τρόπο κρυπτογράφησης και επεξεργασίας του τελευταίου μπλοκ, και τη μέθοδο που υιοθετείται για τη δημιουργία της τελικής εξόδου (μετα-επεξεργασία). Ο τρόπος κρυπτογράφησης και επεξεργασίας του τελευταίου μπλοκ και οι μέθοδοι μετα-επεξεργασίας επηρεάζουν τον αριθμό των κρυπτογραφικών κλειδιών που απαιτούνται για τη δημιουργία ενός MAC.

Στην απλούστερη μορφή του ένα MAC υπολογίζεται ως το τελευταίο μπλοκ κρυπτοκειμένου που υπολογίσθηκε με τρόπο λειτουργίας CBC. Αυτό είναι το λεγόμενο CBC-MAC, το οποίο ωστόσο στη βασική του

¹Οι «επιθέσεις σύγκρουσης» στα MAC αναφέρονται σε κακόβουλες προσπάθειες όπου ο επιτιθέμενος προσπαθεί να δημιουργήσει δύο διαφορετικά μηνύματα που παράγουν το ίδιο MAC. Αν ένας κρυπτογραφικός αλγόριθμος MAC είναι ευάλωτος σε επιθέσεις σύγκρουσης, αυτό θα μπορούσε να επιτρέψει σε έναν επιτιθέμενο να δημιουργήσει παραπλανητικά μηνύματα που θα επαληθεύονται ως γνήσια από τον παραλήπτη. Για να προστατευθούμε από αυτούς τους τύπους επιθέσεων, είναι σημαντικό να χρησιμοποιούνται ισχυροί αλγόριθμοι MAC που δεν είναι ευάλωτοι σε γνωστές μεθόδους επιθέσεων σύγκρουσης και να τηρούνται βέλτιστες πρακτικές για τη διαχείριση των κλειδιών.

μορφή δε θεωρείται ασφαλές [2], εκτός από ορισμένες περιπτώσεις, όπως για παράδειγμα όταν το μήκος του αρχικού μηνύματος προσαρτάται σε αυτό πριν τον υπολογισμό του MAC.

Το πρότυπο ISO 9797-1 [3] ορίζει τέσσερις μεθόδους πλήρωσης, τρεις μεθόδους κρυπτογράφησης και επεξεργασίας του τελευταίου μπλοκ, και τρεις μεθόδους μετα-επεξεργασίας της εξόδου. Επιπλέον, ορίζει έξι αλγορίθμους CBC-MAC που μπορούν να χρησιμοποιηθούν με οποιοδήποτε αλγόριθμο κρυπτογράφησης. Σε αυτό το Κεφάλαιο δίνεται μια συνοπτική περιγραφή των αλγορίθμων, ενώ έμφαση δίνεται στον CMAC που αποτελεί έναν από τους εγκεκριμένους αλγορίθμους MAC του NIST [2], μαζί με τον HMAC που αναλύεται στην ενότητα 5.2.2 και τον KMAC (Keccak Message Authentication Code) [4] που βασίζεται στην συνάρτηση σύνοψης Keccak.

Ο Πίνακας 5.2 συνοψίζει αυτούς τους έξι αλγορίθμους, όπου H_q είναι η έξοδος της κρυπτογράφησης και επεξεργασίας του τελευταίου μπλοκ, ή αλλιώς, τελικής επανάληψης, H_{q-1} είναι η έξοδος της προτελευταίας επανάληψης, D_i είναι το i -οστό μπλοκ του μηνύματος με πλήρωση, και K είναι το κρυπτογραφικό κλειδί για τον κρυπτογραφικό αλγόριθμο μπλοκ που χρησιμοποιείται για τις επαναλήψεις $1, \dots, q - 1$. Σε σχήματα που χρησιμοποιούν πρόσθετα κλειδιά, όπως για παράδειγμα τα K' και K'' , όλα τα κλειδιά προέρχονται από ένα μόνο κλειδί με τρόπο που καθορίζεται από το πρότυπο.

Συνοπτικά, στους βασικούς κώδικες αυθεντικοποίησης μηνύματος που ορίζονται στο πρότυπο ISO-9797-1 [3] περιλαμβάνονται οι ακόλουθοι (βλέπε Πίνακα 5.2):

- CBC-MAC: το MAC είναι το τελευταίο μπλοκ κρυπτοκειμένου που υπολογίσθηκε με τρόπο λειτουργίας CBC.
- EMAC (Encrypted CBC-MAC): το MAC είναι το αποτέλεσμα της επανακρυπτογράφησης του CBC-MAC με ένα διαφορετικό κλειδί K' .
- AMAC (ANSI Retail MAC): το MAC είναι το αποτέλεσμα της αποκρυπτογράφησης του τελευταίου μπλοκ κρυπτοκειμένου με ένα διαφορετικό κλειδί K' και της εκ νέου κρυπτογράφησης του αποτέλεσματος με το κλειδί K .
- CMAC (Cipher-based MAC): Όπως και με το CBC-MAC, το MAC είναι το τελευταίο μπλοκ κρυπτοκειμένου που υπολογίσθηκε με τρόπο λειτουργίας CBC. Η διαφορά των δύο σχημάτων αφορά στον τρόπο με τον οποίο γίνεται η κρυπτογράφηση στην τελευταία επανάληψη. Ορίζεται επίσης στο NIST SP 800-38b [2].
- LMAC (Linear MAC): Όπως και με το CBC-MAC μόνο που στην τελευταία επανάληψη η κρυπτογράφηση γίνεται με ένα διαφορετικό κλειδί K' .

Γενικά, τα EMAC, AMAC και CMAC θεωρούνται τα πιο συχνά χρησιμοποιούμενα σχήματα. Το μέγεθος του κλειδιού K θα πρέπει να καθορίζεται από το απαιτούμενο επίπεδο ασφάλειας (βλέπε Ορισμό 1.1 στην Ενότητα 1.5), με παρόμοιο τρόπο όπως και με την περίπτωση ενός κρυπτογραφικού αλγορίθμου μπλοκ. Δηλαδή, για ένα επίπεδο ασφαλείας k bit, το μέγεθος του κλειδιού θα πρέπει ιδανικά να επιλέγεται ίσο με k bit. Το μέγεθος εξόδου s καθορίζεται από τον αριθμό των προσπαθειών επαλήθευσης MAC που μπορούν να γίνουν. Ο αριθμός αυτός περιορίζεται από τη διάρκεια ζωής του κρυπτοσυστήματος και το χρόνο που απαιτείται για την επαλήθευση μιας τιμής MAC. Σε κάθε περίπτωση, για έναν καλά σχεδιασμένο MAC, η πιθανότητα επιτυχίας μιας προσπάθειας επαλήθευσης του MAC θα πρέπει να είναι 2^{-s} . Για παράδειγμα, αν το μέγεθος εξόδου s έχει επιλεγεί να είναι 128 bits, η πιθανότητα επιτυχίας μιας προσπάθειας επαλήθευσης του MAC από έναν επιτιθέμενο, χωρίς να γνωρίζει το μυστικό κλειδί, θα πρέπει να είναι 1 στις 2^{128} .

5.2.1.1 EMAC

Το σχήμα EMAC (ή Encrypted CBC-MAC) προβλέπει την επανακρυπτογράφηση με ένα διαφορετικό κλειδί (έστω K'), του τελευταίου μπλοκ κρυπτοκειμένου που υπολογίσθηκε με τρόπο λειτουργίας CBC και κλειδί

Πίνακας 5.2: Οι συναρτήσεις MAC σύμφωνα με το ISO 9797-1.

Σχήμα κατά ISO 9797-1	Πρώτη Επανάληψη	Τελευταία Επανάληψη	Μετά-Επεξεργασία	Γνωστό και ως
1	$H_1 = E_K(D_1)$	$H_q = E_K(D_q \oplus H_{q-1})$	$G = H_q$	CBC-MAC
2	$H_1 = E_K(D_1)$	$H_q = E_K(D_q \oplus H_{q-1})$	$G = E_{K'}(H_q)$	EMAC
3	$H_1 = E_K(D_1)$	$H_q = E_K(D_q \oplus H_{q-1})$	$G = E_K(D_{K'}(H_q))$	AMAC
4	$H_1 = E_{K''}(E_K(D_1))$	$H_q = E_K(D_q \oplus H_{q-1})$	$G = E_{K'}(H_q)$	-
5	$H_1 = E_K(D_1)$	$H_q = E_K(D_q \oplus H_{q-1} \oplus K')$	$G = H_q$	CMAC
6	$H_1 = E_K(D_1)$	$H_q = E_{K'}(D_q \oplus H_{q-1})$	$G = H_q$	LMAC

Κ. Προτάθηκε το 2000 [5] και ορίζεται ως Αλγόριθμος 2 στο ISO-9797-1. Υπάρχουν γνωστές επιθέσεις κατά του μηχανισμού που απαιτούν $2^{n/2}$ επαληθεύσεις MAC, όπου n το μέγεθος του μπλοκ. Παραλλαγή του σχήματος με αποδεδειγμένα επίπεδα ασφάλειας, όπου χρησιμοποιούνται δύο ανεξάρτητα κλειδιά έχει προταθεί στα [5, 6].

5.2.1.2 AMAC

Το σχήμα AMAC (ή ANSI Retail MAC) προβλέπει την αποκρυπτογράφηση του τελευταίου μπλοκ κρυπτοειδένευν με ένα διαφορετικό κλειδί K' από το κλειδί K που χρησιμοποιήθηκε αρχικά για την κρυπτογράφηση του μηνύματος με τρόπο λειτουργίας CBC, καθώς και την εκ νέου κρυπτογράφηση του αποτελέσματος με το αρχικό κλειδί K . Ορίζεται ως Αλγόριθμος 3 στο ISO 9797-1. Ο αλγόριθμος είναι γνωστός ως ANSI Retail MAC, λόγω του ότι αρχικά είχε προταθεί στο πρότυπο ANSI X9.19 [7], ή απλώς AMAC για συντομία. Χρησιμοποιείται σε τραπεζικές εφαρμογές με τον DES ως τον υποκείμενο αλγόριθμο κρυπτογράφησης. Ένα μειονέκτημα του AMAC είναι ότι μια εσωτερική σύγκρουση (collision) επιτρέπει τη δημιουργία πλαστογραφημένων MAC αλλά και την αποτελεσματική ανάκτηση κλειδιού. Μια επίθεση εσωτερικής σύγκρουσης είναι ένας τύπος επίθεσης όπου ένας μη εξουσιοδοτημένος χρήστης προσπαθεί να βρει δύο μηνύματα που παράγουν το ίδιο MAC. Αυτό επιτυγχάνεται με την εκμετάλλευση αδυναμιών στον αλγόριθμο MAC ή/και στην υποκείμενη συνάρτηση σύνοψης.

5.2.1.3 CMAC

Το σχήμα CMAC (ή Cipher-based MAC) προτάθηκε στο [8] και τυποποιήθηκε ως Αλγόριθμος 5 στο ISO 9797-1 [3]. Αποτελεί έναν από τους εγκεκριμένους αλγορίθμους MAC του NIST [2]. Τα βήματα του αλγορίθμου αποτυπώνονται στον παρακάτω Αλγόριθμο 5.1 [2].

Στο βήμα 1, τα δευτερεύοντα κλειδιά δημιουργούνται από το κλειδί K . Στα βήματα 2-11, το μήνυμα εισόδου μορφοποιείται σε μια ακολουθία από ακέραια μπλοκ όπου το τελευταίο μπλοκ προκύπτει από την εφαρμογή της πράξης XOR με εισόδους το τρέχον περιεχόμενό του και ένα δευτερεύον κλειδί. Υπάρχουν δύο περιπτώσεις οι οποίες απεικονίζονται στο Σχήμα 5.1:

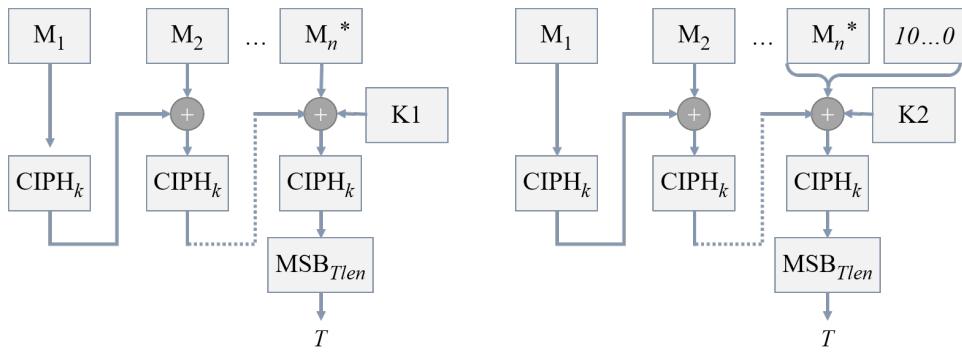
- Εάν το μήκος του μηνύματος είναι πολλαπλάσιο του μήκους του μπλοκ, τότε το μήνυμα χωρίζεται σε έναν ακέραιο αριθμό από μπλοκ. Το τελευταίο μπλοκ προκύπτει από την εφαρμογή της πράξης XOR με εισόδους το τρέχον περιεχόμενό του και το πρώτο δευτερεύον κλειδί.
- Εάν το μήκος του μηνύματος δεν είναι ακέραιο πολλαπλάσιο του μήκους του μπλοκ, τότε το μήνυμα διαιρείται σε μια ακολουθία μπλοκ ακολουθούμενη από μια ακολουθία από bits της οποίας το μήκος είναι μικρότερο από το μήκος του μπλοκ. Σε αυτήν την ακολουθία bits προστίθεται μια ακολουθία από bit πλήρωσης (padding). Συγκεκριμένα, προστίθεται ένα bit με τιμή “1” ακολουθούμενο από τόσα bit

Αλγόριθμος 5.1: Διαδικασία δημιουργίας CMAC.

Είσοδος: Μήνυμα M μήκους $Mlen$ (σε bit);
Έξοδος: MAC T μήκους $Tlen$ bit;

- 1 Δημιουργία δευτερευόντων κλειδιών (σύμφωνα με τον παρακάτω Αλγόριθμο 5.2) για τα κλειδιά K_1 και K_2 .
- 2 **if** $Mlen = 0$ **then**
- 3 2 | $n = 1$
- 4 3 **else**
- 5 4 | $n = \lceil Mlen/b \rceil // \lceil x \rceil$ είναι ο ελάχιστος ακέραιος που δεν είναι μικρότερος από τον πραγματικό αριθμό x και b το μήκος του μπλοκ σε bits.
- 6 5 **end**
- 7 6 'Εστω $M_1, M_2, \dots, M_{n-1}, M_n^*$ η μοναδική ακολουθία συμβολοσειρών bit τέτοια ώστε $M = M_1 \parallel M_2 \parallel \dots \parallel M_{n-1} \parallel M_n^*$ όπου M_1, M_2, \dots, M_{n-1} ακέραια μπλοκ και \parallel η συνένωσή τους (εάν $Mlen \leq b$ τότε $M = M_1^*$).
- 8 7 **if** M_n^* είναι ένα ακέραιο μπλοκ **then**
- 9 8 | $M_n = K1 \oplus M_n^*$
- 10 9 **else**
- 11 10 | $M_n = K2 \oplus M_n^* \parallel 10^j$, όπου $j = nb - Mlen - 1$
- 12 11 **end**
- 13 12 $C_0 = 0^b$
- 14 13 **for** $i \leftarrow 0$ **to** n **do**
- 15 14 | $C_i = E_K(C_{i-1} \oplus M_i)$ // Υπολογισμός μπλοκ κρυπτογράφησης με κλειδί K .
- 16 15 **end**
- 17 16 $T = MSB_{Tlen}(C_n)$ // Επέλεξε από το C_n τα αριστερότερα $Tlen$ bit.

με τιμή “0”, ενδεχομένως και κανένα, όσα είναι απαραίτητα για να σχηματιστεί ένα πλήρες μπλοκ. Το τελευταίο μπλοκ προκύπτει από την εφαρμογή της πράξης XOR με εισόδους το τρέχον περιεχόμενό του και το δεύτερο δευτερεύον κλειδί.



Σχήμα 5.1: Οι δύο περιπτώσεις δημιουργίας του CMAC.

Στα βήματα 12-15, γίνεται η κρυπτογράφηση του κάθε μπλοκ με τον τρόπο λειτουργίας CBC, χρησιμοποιώντας ως διάνυσμα αρχικοποίησης το μηδενικό μπλοκ. Στα βήματα 16 και 17, το τελευταίο μπλοκ ιρυπτοκειμένου περικόπτεται (truncated) στο προβλεπόμενο μήκος και το αποτέλεσμα επιστρέφεται ως MAC.

Η δημιουργία των κλειδιών K_1 και K_2 γίνεται σύμφωνα με τον παρακάτω Αλγόριθμο 5.2.

Τα επίπεδα ασφάλειας που προσφέρει ο μηχανισμός CMAC έχουν αποδειχτεί για την περίπτωση που ο υποκειμενος κρυπτογραφικός αλγόριθμος μπλοκ παρέχει μια ψευδοτυχαία αντιμετάθεση [9]. Όταν χρησι-

Αλγόριθμος 5.2: Δημιουργία δευτερεύοντων κλειδιών του CMAC.

Είσοδος: Κλειδί K ;

Έξοδος: Δευτερεύοντα κλειδιά $K1$ και $K2$;

- 1 $L = E_K(0^b)$ // όπου b το μέγεθος του μπλοκ του αλγορίθμου κρυπτογράφησης.
- 2 **if** $MSB_1(L) = 0$ **then**
- 3 $K1 = L \ll 1$
- 4 **else**
- 5 $K1 = (L \ll 1) \oplus Rb$ // Το Rb είναι μια συμβολοσειρά bits που καθορίζεται πλήρως από τον αριθμό των bits σε ένα μπλοκ. Συγκεκριμένα, για μπλοκ μήκους 128 και 64 bit: $R_{128} = 0^{120}10000111$ και $R_{64} = 0^{59}11011$.
- 6 **end**
- 7 **if** $MSB_1(K1) = 0$ **then**
- 8 $K2 = K1 \ll 1$
- 9 **else**
- 10 $K2 = (K1 \ll 1) \oplus Rb$
- 11 **end**

μοποιείται ο AES-128, τα υπάρχοντα πρότυπα συνιστούν ότι το κρυπτοσύστημα που κάνει χρήση CMAC θα πρέπει να χρησιμοποιείται για το πολύ 2^{48} μηνύματα. Μετά από 2^{48} μηνύματα, η πιθανότητα εσωτερικής σύγκρουσης είναι 2^{-32} . Εάν εντοπιστεί μια τέτοια σύγκρουση, η πλαστογράφηση MAC, δηλαδή η μη εξουσιοδοτημένη δημιουργία ενός έγκυρου MAC, καθίσταται εφικτή.

5.2.2 MAC Βασισμένα σε Συναρτήσεις Σύνοψης

Τα MAC αυτού του τύπου απαιτούν τη χρήση μιας συνάρτησης σύνοψης σε συνδυασμό με ένα κλειδί. Το HMAC (Keyed-Hashing for Message Authentication) που ορίζεται στο ISO/IEC 9797-2 [10] καθώς και στα RFC 2104 [11] και FIPS 198 [12], είναι ένα παράδειγμα MAC αυτού του τύπου.

Η βασική ιδέα πίσω από το HMAC είναι η δημιουργία σύνοψης με μια συνάρτηση σύνοψης H (π.χ. SHA-256 ή SHA-3) συνδυαστικά με τη χρήση ενός μυστικού κλειδιού K . Το αποτέλεσμα είναι ένα MAC που είναι μοναδικό για το μήνυμα και το κλειδί, και μπορεί να χρησιμοποιηθεί για την επαλήθευση της ακεραιότητας (λόγω της σύνοψης) και την αυθεντικοποίηση (λόγω του μυστικού κλειδιού K) του μηνύματος. Υποθέτουμε ότι η H κατακερματίζει τα δεδομένα επαναλαμβάνοντας μια βασική συνάρτηση συμπίεσης σε κάθε μπλοκ του μηνύματος. Έστω B το μήκος σε bytes τέτοιων μπλοκ (ενδεικτικά μήκη μπλοκ για τις συναρτήσεις σύνοψης, όπως αυτές των οικογενειών SHA-2 και SHA-3, είναι τα 64 και 128 bytes) και L το μήκος σε bytes της σύνοψης του μηνύματος (για τις συναρτήσεις των οικογενειών SHA-2, SHA-3 αυτό κυμαίνεται από 28 έως 64 bytes). Το κλειδί K μπορεί να είναι οποιουδήποτε μήκους μέχρι B byte, όπου B το μήκος του μπλοκ της συνάρτησης σύνοψης. Οι εφαρμογές παραγωγής HMAC που χρησιμοποιούν κλειδιά μεγαλύτερα από B bytes θα κατακερματίσουν πρώτα το κλειδί χρησιμοποιώντας τη συνάρτηση σύνοψης H και στη συνέχεια θα χρησιμοποιήσουν το αποτέλεσμα των L bytes ως το πραγματικό κλειδί στο HMAC. Σε κάθε περίπτωση, το ελάχιστο συνιστώμενο μήκος για το K είναι L bytes.

Το HMAC για το μήνυμα m , με τη χρήση του κλειδιού K , ορίζεται σύμφωνα με το RFC 2104 [11] ως εξής (η δημιουργία του περιγράφεται αναλυτικά στον Αλγόριθμο 5.3):

$$HMAC(K, m) = H((K \oplus opad) \parallel H((K \oplus ipad) \parallel m))$$

όπου

opad: εξωτερική πλήρωση μήκους ενός μπλοκ, αποτελούμενη από επαναλαμβανόμενα bytes με τιμή $0x5c$

ipad: εσωτερική πλήρωση μήκους ενός μπλοκ, αποτελούμενη από επαναλαμβανόμενα bytes με τιμή 0x36

Αλγόριθμος 5.3: Διαδικασία δημιουργίας HMAC.

Είσοδος: Μήνυμα m ;
Κλειδί K ;

Έξοδος: HMAC;

- 1 **if** $length(K) > B$ **then**
- 2 $K = H(K)$ // Κλειδιά με μήκος μεγαλύτερο του μήκους του μπλοκ (B), θα πρέπει προηγουμένως να κατακερματιστούν.
- 3 **else**
- 4 $K = K \parallel zeroes(B - length(K))$ // Στα κλειδιά με μήκος μικρότερο του μήκους του μπλοκ (B), θα πρέπει να γίνει πλήρωση με μηδενικά -- το σύμβολο \parallel δηλώνει συνένωση.
- 5 **end**
- 6 $opad = [0x5c * B]$
- 7 $ipad = [0x36 * B]$
- 8 $HMAC = H(K \oplus opad \parallel H(K \oplus ipad \parallel m))$

Το HMAC τυπικά μπορεί να χρησιμοποιηθεί με οποιαδήποτε υποκείμενη συνάρτηση σύνοψης όπως οι SHA-2 και SHA-3. Το HMAC που βασίζεται στη χρήση του MD4, γνωστό και ως HMAC-MD4, δεν πρέπει να χρησιμοποιείται ενώ τα HMAC-SHA1 και HMAC-MD5 εξακολουθούν να είναι ανθεκτικά σε πλαστογραφίες. Ωστόσο, η χρήση του MD5 δεν προτείνεται.

Το HMAC έχει σχεδιαστεί για να είναι ανθεκτικό σε διάφορους τύπους επιθέσεων, συμπεριλαμβανομένων των επιθέσεων σύγκρουσης και των επιθέσεων γενεθλίων. Χρησιμοποιείται ευρέως σε μια ποικιλία εφαρμογών, όπως πρωτόκολλα προστασίας των επικοινωνιών (π.χ. SSL/TLS), ψηφιακές υπογραφές και ενημερώσεις λογισμικού.

Ένα πλεονέκτημα του HMAC είναι ότι είναι σχετικά απλό στη λειτουργία του και μπορεί να χρησιμοποιηθεί σε ένα ευρύ φάσμα εφαρμογών κρυπτογραφικών συναρτήσεων κατακερματισμού. Επιπλέον, το HMAC είναι συχνά ταχύτερο από άλλους τύπους MAC, όπως το CBC-MAC, ενώ είναι λιγότερο επιρρεπές σε ορισμένους τύπους επιθέσεων, όπως επιθέσεις επέκτασης μήκους (length extension attacks), όπως αυτές αναλύονται στην Ενότητα 5.2.2.1.

5.2.2.1 Επίθεση Επέκτασης Μήκους

Η Επίθεση Επέκτασης Μήκους (Length Extension Attack) είναι μια κρυπτογραφική επίθεση που εκμεταλλεύεται τις ιδιότητες ορισμένων συναρτήσεων σύνοψης. Σε αυτήν την επίθεση, ο επιτιθέμενος μπορεί να προσθέσει επιπλέον δεδομένα στο τέλος ενός κατακερματισμένου μηνύματος χωρίς να γνωρίζει το αρχικό μήνυμα ή το μυστικό κλειδί. Αυτό μπορεί να επιτευχθεί επειδή οι περισσότερες συναρτήσεις κατακερματισμού, όπως οι MD5 και SHA-1, εσωτερικά βασίζονται στην προσάρτηση δεδομένων κατά τη διαδικασία κατακερματισμού [13]. Μια τέτοια επίθεση μπορεί να έχει σοβαρές επιπτώσεις για τα συστήματα που βασίζονται στην ακεραιότητα και την αυθεντικότητα των δεδομένων μέσω κατακερματισμού. Για παράδειγμα, αν ένας επιτιθέμενος καταφέρει να αποκτήσει μια έγκυρη σύνοψη ενός μηνύματος, μπορεί να δημιουργήσει έναν μια νέα έγκυρη σύνοψη για το μήνυμα με πρόσθετα δεδομένα, παρακάμπτοντας τους μηχανισμούς ασφαλείας.

Όταν, για παράδειγμα, χρησιμοποιείται η δομή της απλής συνένωσης μεταξύ του κλειδιού K και του μηνύματος m , δηλαδή $Hash(K \parallel m)$, και το μήκος του μηνύματος και του κλειδιού είναι γνωστά, μια επίθεση επέκτασης μήκους επιτρέπει σε οποιονδήποτε να συμπεριλάβει επιπλέον πληροφορίες στο τέλος του μηνύματος και να δημιουργήσει ένα έγκυρο MAC χωρίς να γνωρίζει το κλειδί. Έτσι, σε μια επίθεση επέκτασης μήκους

ο επιτιθέμενος μπορεί να χρησιμοποιήσει το $Hash(m1)$ και το μήκος του μηνύματος $m1$ για να υπολογίσει το $Hash(m1 \parallel m2)$ για ένα μήνυμα $m2$ που ελέγχεται από τον επιτιθέμενο, χωρίς να χρειάζεται να γνωρίζει το περιεχόμενο του $m1$.

Η ευπάθεια των αλγορίθμων MAC σε επιθέσεις επέκτασης μήκους εξαρτάται από τον συγκεκριμένο αλγόριθμο και το μήκος της εσωτερικής κατάστασης που χρησιμοποιείται από τη συνάρτηση σύνοψης. Ορισμένοι αλγόριθμοι MAC, όπως ο HMAC-SHA256, έχουν σχεδιαστεί για να είναι ανθεκτικοί σε επιθέσεις επέκτασης μήκους ενσωματώνοντας μια μυστική τιμή βασισμένη στο κλειδί στη συνάρτηση σύνοψης, ενώ άλλοι, όπως η απλή συνένωση ή το CBC-MAC, είναι ευάλωτοι σε επιθέσεις επέκτασης μήκους.

5.2.3 Μέγεθος MAC

Οι αλγόριθμοι MAC συνήθως εξάγουν ένα μπλοκ δεδομένων το οποίο στη συνέχεια περικόπτεται για να σχηματιστεί ένα MAC μικρότερου μήκους. Παράδειγμα αποτελεί ο αλγόριθμος CBC-MAC που περικόπτει την έξοδο από 8 bytes σε 4 bytes ή σε ορισμένες περιπτώσεις ακόμη και σε 3 bytes. Αυτή η προσέγγιση μειώνει σαφώς τον όγκο των δεδομένων που πρέπει να σταλούν στον παραλήπτη, αλλά σε ορισμένες περιπτώσεις μπορεί επιπλέον να αυξήσει την ασφάλεια, καθώς παρέχει σε έναν επίδοξο επιτιθέμενο λιγότερες πληροφορίες για να εργαστεί.

Ωστόσο, ένα επιχείρημα κατά της χρήσης σύντομων MAC είναι ότι με αυτά τα MAC αυξάνονται και οι πιθανότητες ενός επιτιθέμενου να μαντέψει την τιμή τους [2]. Ένα MAC με επίπεδο ασφάλειας 2^s θα πρέπει να έχει μέγεθος κλειδιού τουλάχιστον s bits και μέγεθος εξόδου τουλάχιστον s bits.

5.3 Αυθεντικοποιημένη Κρυπτογράφηση

Οι μηχανισμοί Αυθεντικοποιημένης Κρυπτογράφησης (Authenticated Encryption) ικανοποιούν όλες τις απαιτήσεις σχετικά με την προστασία δεδομένων κατά τη διακίνησή τους, δηλαδή την εμπιστευτικότητα και την ακεραιότητα μηνύματος, καθώς και την αυθεντικοποίηση της πηγής. Υπάρχουν δύο κατηγορίες μηχανισμών που έχουν αυτές τις ιδιότητες: οι μηχανισμοί γενικής χρήσης, και οι αποκλειστικοί μηχανισμοί, δηλαδή μηχανισμοί που σχεδιάζονται αποκλειστικά για αυτόν τον σκοπό.

Η αυθεντικοποιημένη κρυπτογράφηση με συσχετισμένα δεδομένα (Authenticated Encryption with Associated Data – AEAD) είναι μια επέκταση της αυθεντικοποιημένης κρυπτογράφησης που επιτρέπει στον αποστολέα να εισάγει σε ένα μήνυμα δεδομένα που πρέπει να αυθεντικοποιηθούν, αλλά όχι να κρυπτογραφηθούν, όπως είναι τα δεδομένα κάποιας επικεφαλίδας. Όλοι οι τρόποι που περιγράφονται σε αυτήν την ενότητα είναι σχήματα AEAD, με εξαίρεση τους μηχανισμούς γενικής χρήσης που εξαρτώνται από τις λεπτομέρειες της υλοποίησης. Αρχικά περιγράφουμε τους μηχανισμούς γενικής χρήσης, οι οποίοι συνήθως χρησιμοποιούν δύο ανεξάρτητα κλειδιά, ένα για την κρυπτογράφηση και ένα για το MAC. Τα σχήματα AEAD έχουν ένα μόνο κλειδί και για τις δύο λειτουργίες.

Ο Πίνακας 5.3 παρουσιάζει το σύνολο των μηχανισμών που χρησιμοποιούνται για αυθεντικοποιημένη κρυπτογράφηση και αναλύονται στις επόμενες ενότητες.

5.3.1 Μηχανισμοί Γενικής Χρήσης

Οι μηχανισμοί γενικής χρήσης συνδυάζουν κρυπτογραφικούς μηχανισμούς, όπως κρυπτογράφηση και ψηφιακές υπογραφές ή κρυπτογράφηση και MAC. Ως αποτέλεσμα, υπάρχουν οι παρακάτω επιλογές:

- Κρυπτογράφηση και ψηφιακή υπογραφή. Οι δύο μηχανισμοί μπορούν να συνδυαστούν σύμφωνα με τους ακόλουθους τρόπους: *Κρυπτογράφηση-και-μετά-υπογραφή* (encrypt-then-sign), και *Υπογραφή-και-μετά-κρυπτογράφηση* (sign-then-encrypt).

Τόσο ο μηχανισμός *Κρυπτογράφηση-και-μετά-υπογραφή* όσο και ο *Υπογραφή-και-μετά-κρυπτογράφηση* παρέχουν επαρκή προστασία για την ασφάλεια των δεδομένων και η επιλογή μεταξύ των δύο εξαρτάται

Πίνακας 5.3: Αυθεντικοποιημένη κρυπτογράφηση.

Μηχανισμός	Κατηγοριοποίηση Παλαιού Τύπου	Κατηγοριοποίηση Μελλοντική Χρήση	Σημειώσεις
Γενική χρήση	✓	✗	Κρυπτογράφηση-και-μετά-MAC και άλλες παραλλαγές
CCM	✓	✗	Αντικαταστάθηκε από το EAX
CWC	✓	✗	Αντικαταστάθηκε από το GCM
OCB	✓	✓	
EAX	✓	✓	
GCM	✓	✓	
ChaCha20+Poly1305	✓	✓	

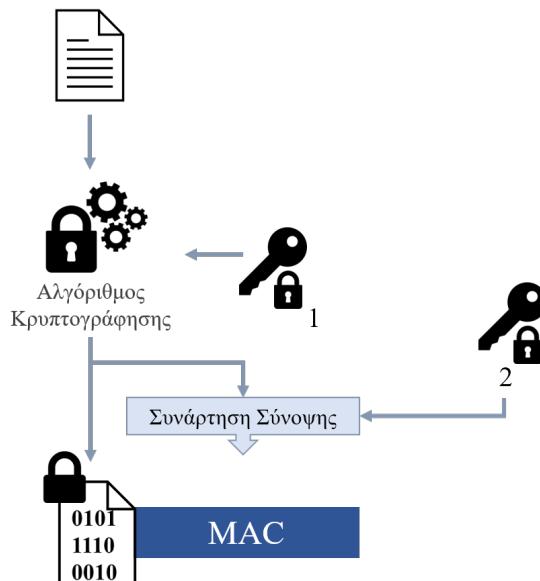
από τη συγκεκριμένη περίπτωση χρήσης και το επιθυμητό επίπεδο ασφάλειας. Με τον μηχανισμό *Κρυπτογράφηση-και-μετά-υπογραφή* τα δεδομένα κρυπτογραφούνται πρώτα χρησιμοποιώντας ένα συμμετρικό κλειδί και στη συνέχεια (τα κρυπτογραφημένα δεδομένα) υπογράφονται χρησιμοποιώντας το ιδιωτικό κλειδί του υπογράφοντος. Αυτός ο τρόπος παρέχει εμπιστευτικότητα και ακεραιότητα για τα δεδομένα, καθώς η κρυπτογράφηση διασφαλίζει ότι μόνο εξουσιοδοτημένα μέρη μπορούν να διαβάσουν τα δεδομένα, ενώ η υπογραφή διασφαλίζει την αυθεντικοποίηση πηγής, καθώς και ότι τα δεδομένα δεν έχουν τροποποιηθεί. Στον μηχανισμό *Υπογραφή-και-μετά-κρυπτογράφηση* τα δεδομένα αρχικά υπογράφονται και στη συνέχεια, η υπογραφή και τα δεδομένα κρυπτογραφούνται χρησιμοποιώντας ένα συμμετρικό κλειδί. Γενικά ο μηχανισμός *Υπογραφή-και-μετά-κρυπτογράφηση* θεωρείται πιο ασφαλής από τον *Κρυπτογράφηση-και-μετά-υπογραφή*, καθώς αποτρέπει κακόβουλες ενέργειες όπου ο επιτιθέμενος αντικαθιστά την υπογραφή του αποστολέα με τη δική του, προσποιούμενος πως αυτός είναι η πηγή του μηνύματος, το οποίο όμως δεν γνωρίζει καθώς είναι κρυπτογραφημένο.

- Κρυπτογράφηση και MAC (δύο διαφορετικά κλειδιά). Οι δύο μηχανισμοί μπορούν να συνδυαστούν σύμφωνα με τους ακόλουθους τρόπους:
 - Κρυπτογράφηση και μετά MAC (συνιστάται) (βλέπε Σχήμα 5.2)
 - Κρυπτογράφηση και MAC (δεν συνιστάται)
 - MAC και μετά κρυπτογράφηση (δεν συνιστάται)

Οι διάφοροι τρόποι συνδυασμού κρυπτογράφησης και MAC συζητήθηκαν από τους Bellare και Nam-prempre [14], συμπεριλαμβανομένου του τρόπου *Κρυπτογράφηση-και-μετά-MAC*. Το συμπέρασμά τους είναι ότι το *Κρυπτογράφηση-και-μετά-MAC* είναι ο μόνος τρόπος με τον οποίο μπορεί κανείς να συνδυάσει αποτελεσματικά ένα σχήμα κρυπτογράφησης με ένα MAC και να επιτύχει ασφάλεια. Τα υπόλοιπα σχετικά σχήματα, όπως το *Κρυπτογράφηση-και-μετά-MAC* ή το *MAC-και-μετά-Κρυπτογράφηση*, γενικά δεν θα πρέπει να χρησιμοποιούνται καθώς διάφορες επιθέσεις έχουν πραγματοποιηθεί σε συστήματα που χρησιμοποιούν αυτές τις μη ασφαλείς παραλλαγές, όπως στο TLS [15].

5.3.2 OCB

Το OCB (Offset Codebook) [16] είναι ένας τρόπος λειτουργίας κρυπτογραφικών αλγορίθμων μπλοκ, που παρέχει αυθεντικοποιημένη κρυπτογράφηση με δυνατότητα συσχέτισης δεδομένων (AEAD). Προτάθηκε από τους Phillip Rogaway και Mihir Bellare το 2003 [17] και θεωρείται ένας από τους πιο αποτελεσματικούς τρόπους λειτουργίας AEAD από άποψη απόδοσης κυρίως λόγω του γεγονότος ότι είναι μονού-περάσματος.



Σχήμα 5.2: Αυθεντικοποιημένη κρυπτογράφηση με την μέθοδο κρυπτογράφηση και μετά MAC.

Απαιτείται, δηλαδή, μόνο μία κλήση εκτέλεσης του κρυπτογραφικού αλγορίθμου για κάθε μπλοκ αρχικού κειμένου, ενώ απαιτούνται δύο επιπλέον κλήσεις του κρυπτογραφικού αλγορίθμου για την ολοκλήρωση της διαδικασίας κρυπτογράφησης. Επιπλέον, είναι αποδεδειγμένα ασφαλής, θεωρώντας ότι ο υποκείμενος κρυπτογραφικός αλγόριθμος μπλοκ ασφαλής.

Αφού το σχήμα OCB χρησιμοποιεί κρυπτογραφικούς αλγορίθμους μπλοκ για την κρυπτογράφηση και τον έλεγχο αυθεντικοποίησης δεδομένων ακολουθείται όλη η τυπική διαδικασία χρήσης αυτών των κρυπτογραφικών αλγορίθμων, με τη διαίρεση του αρχικού κειμένου σε μπλοκ σταθερού μήκους και την πλήρωση του περιεχομένου του τελευταίου μπλοκ εάν είναι απαραίτητο. Στη συνέχεια δημιουργείται μια τυχαία και μοναδική τιμή (nonce), η οποία στη συνέχεια χρησιμοποιείται ως είσοδος σε έναν κρυπτογραφικό αλγόριθμο μπλοκ σε λειτουργία μετρητή (counter code) ώστε από την έξοδό του να προκύπτει μια κλειδοροή από τη μοναδική τιμή (nonce) χρησιμοποιώντας τον κρυπτογραφικό αλγόριθμο μπλοκ σε λειτουργία μετρητή (counter mode). Το κρυπτογραφημένο κείμενο αποτελεί το αποτέλεσμα της συνάρτησης αποκλειστικού-Η (XOR) της κλειδοροής με το αρχικό κείμενο.

Εκτός από την κρυπτογράφηση του απλού κειμένου, ο τρόπος λειτουργίας OCB παρέχει επίσης έλεγχο ταυτότητας πηγής μηνυμάτων. Το OCB υπολογίζει έναν κώδικα αυθεντικοποίησης μηνύματος (MAC) για το κρυπτογραφημένο κείμενο και τα συσχετισμένα δεδομένα (εάν υπάρχουν). Το MAC που χρησιμοποιείται στο OCB είναι μια τροποποιημένη έκδοση του κρυπτογραφικού αλγορίθμου που έχει σχεδιαστεί για να παρέχει μια πιο αποτελεσματική και ασφαλή από άλλες συναρτήσεις MAC, όπως το HMAC.

Για την αποκρυπτογράφηση και τον έλεγχο ταυτότητας πηγής ενός μηνύματος, ο παραλήπτης δημιουργεί την ίδια κλειδοροή και την εφαρμόζει στο κρυπτογραφημένο κείμενο για να ανακτήσει το αρχικό κείμενο. Ο παραλήπτης υπολογίζει επίσης το MAC στο κρυπτογραφημένο κείμενο και τα συσχετισμένα δεδομένα χρησιμοποιώντας το ίδιο σχήμα MAC, και το συγκρίνει με το MAC που έλαβε μαζί με το κρυπτογραφημένο μήνυμα. Εάν τα δύο MAC ταιριάζουν, το μήνυμα θεωρείται αυθεντικό.

Το OCB έχει πολλά πλεονεκτήματα σε σχέση με άλλες λειτουργίες AEAD. Παρέχει ισχυρές εγγυήσεις ασφαλείας, συμπεριλαμβανομένης της αντίστασης σε επιθέσεις επιλεγμένου κρυπτογραφημένου κειμένου [18] και επιθέσεις επανάληψης μηνυμάτων. Είναι επίσης εξαιρετικά αποδοτικό από άποψη υπολογιστικού κόστους και κόστους επικοινωνιών, και η λειτουργία του μπορεί να εξελίσσεται παράλληλα ώστε να αξιοποιηθούν οι δυνατότητες των σύγχρονων επεξεργαστών πολλαπλών πυρήνων.

Ωστόσο, η λειτουργία OCB δεν έχει χρησιμοποιηθεί ευρέως στην πράξη λόγω δύο διπλωμάτων ευρεσι-

τεχνίας που είχαν κατοχυρωθεί στις ΗΠΑ, και τα οποία απαιτούν από τους οργανισμούς να καταβάλλουν δικαιώματα για εμπορική χρήση. Από τον Ιανουάριο του 2013, ένας από τους σχεδιαστές δήλωσε ότι η λειτουργία OCB είναι δωρεάν για χρήση λογισμικού υπό μια Γενική Δημόσια Άδεια GNU (GNU General Public License). Ως αποτέλεσμα των αρχικών περιορισμών ωστόσο, άλλες λειτουργίες AEAD, όπως το GCM και το CCM, χρησιμοποιούνται πιο συχνά σε εφαρμογές που απαιτούν αυθεντικοποιημένη κρυπτογράφηση.

Η επέκταση του OCB, γνωστή ως OCB2 [19] δεν πρέπει να χρησιμοποιείται καθώς μια ευπάθεια που ανακαλύφθηκε το 2018 οδήγησε σε πρακτικές και καταστροφικές επιθέσεις, όπως η καθολική πλαστογραφία και η ανάκτηση απλού κειμένου [20].

5.3.3 CCM

Το CCM (Counter with CBC-MAC) τυποποιήθηκε στο NIST SP 800-38c [21] και συνδυάζει τον τρόπο λειτουργίας CTR με το CBC-MAC, χρησιμοποιώντας τον ίδιο κρυπτογραφικό αλγόριθμο μπλοκ και το ίδιο κλειδί. Προορίζεται μόνο για μπλοκ μεγέθους 128-bits και υιοθετείται στο πρότυπο IEEE 802.11i. Η ασφάλεια του CCM μελετήθηκε και αποδείχτηκε από τον Johnson στο [22] ενώ μια σχετική κριτική δόθηκε από τους Rogaway και Wagner στο [23].

Το κύριο μειονέκτημα της λειτουργίας CCM προέρχεται από την μη ικανοποιητική του απόδοση. Για κάθε μπλοκ αρχικού κειμένου απαιτούνται δύο κλήσεις στον κρυπτογραφικό αλγόριθμο. Η λειτουργία CTR επιτρέπει την παράλληλη εκτέλεση, αλλά η λειτουργία CBC-MAC όχι. Επιπλέον, το αρχικό κείμενο θα πρέπει να είναι εξ ολοκλήρου γνωστό πριν να ξεκινήσει η διαδικασία της κρυπτογράφησης, καθώς δεν υποστηρίζει την σύγχρονη (on-the-fly) κρυπτογράφηση αρχικού κειμένου όταν τα αντίστοιχα μπλοκ είναι διαθέσιμα. Αυτός είναι και ένας από τους λόγους που το EAX προτιμάται από το CCM και επομένως το CCM προτείνεται μόνο για εφαρμογές παλαιού τύπου (legacy applications).

5.3.4 EAX

Το EAX [16] προτάθηκε στο [24], όπου παρουσιάστηκε επίσης μια σχετική απόδειξη ασφάλειας. Είναι παρόμοιο με το σχήμα CCM. Επιπλέον, αποτελεί μια μέθοδο δύο περασμάτων που βασίζεται στη λειτουργία CTR και CBC-MAC (χρησιμοποιεί δύο ξεχωριστές διεργασίες για την κρυπτογράφηση και την αυθεντικοποίηση των δεδομένων), αλλά με το πλεονέκτημα ότι τόσο η κρυπτογράφηση όσο και η αποκρυπτογράφηση μπορούν να εκτελεστούν σε πραγματικό χρόνο, δηλαδή ταυτόχρονα κατά την αποστολή και λήψη των δεδομένων, χωρίς καθυστερήσεις.

5.3.5 CWC

Το σχήμα CWC (Carter-Wegman + Counter) σχεδιάστηκε από τους Kohno, Viega και Whiting [25]. Όπως υποδηλώνει το όνομα, συνδυάζει ένα Carter-Wegman MAC για την αυθεντικοποίηση, με κρυπτογράφηση σε τρόπο λειτουργίας CTR για την εμπιστευτικότητα. Είναι αποδεδειγμένα ασφαλές με την προϋπόθεση ότι το διάνυσμα αρχικοποίησης (IV) είναι μια τυχαία τιμή που χρησιμοποιείται μόνο μια φορά (nonce) και ο υποκείμενος κρυπτογραφικός αλγόριθμος μπλοκ είναι ασφαλής. Απαραίτητη προϋπόθεση είναι να μην επαναληφθεί ποτέ το IV, διαφορετικά είναι ευάλωτο σε επιθέσεις πλαστογραφίας. Το NIST επέλεξε να τυποποιήσει το GCM αντί του CWC. Ως αποτέλεσμα, το GCM χρησιμοποιείται και μελετάται πολύ πιο ευρέως ενώ το CWC συνιστάται μόνο για εφαρμογές παλαιού τύπου.

5.3.6 GCM

Το σχήμα GCM (Galois/Counter Mode) σχεδιάστηκε από τους McGrew και Viega [26, 27] και ορίζεται μαζί με την εξειδίκευσή του, το GMAC, στο NIST 800-38D [28]. Το GMAC αφορά την περίπτωση όπου η είσοδος στο GCM δεν απαιτεί κρυπτογράφηση, και επομένως το GCM χρησιμοποιείται ως ένας μηχανισμός αυθεντικοποίησης δεδομένων. Το GCM χρησιμοποιείται ευρέως και συνιστάται ως επιλογή στα RFC IETF

για τα IPsec, SSH και TLS (Κεφάλαιο 10). Επιτρέπει την σύγχρονη (on-the-fly) κρυπτογράφηση, μπορεί εξ ολοκλήρου να υλοποιηθεί με παράλληλη επεξεργασία και ο σχεδιασμός του διευκολύνει αποτελεσματικές υλοποιήσεις σε υλικό.

Ο Αλγόριθμος 5.4 ορίζει τα βήματα που ακολουθούνται για τη δημιουργία της αυθεντικοποιημένης κρυπτογράφησης GCM. Η διαδικασία απεικονίζεται και στο Σχήμα 5.3.

Αλγόριθμος 5.4: Δημιουργία αυθεντικοποιημένης κρυπτογράφησης GCM.

Είσοδος: Διάνυσμα αρχικοποίησης IV;

Αρχικό κείμενο m ;

Πρόσθετα αυθεντικοποιημένα δεδομένα A;

Έξοδος: Κρυπτοκείμενο C;

Ετικέτα αυθεντικοποίησης T;

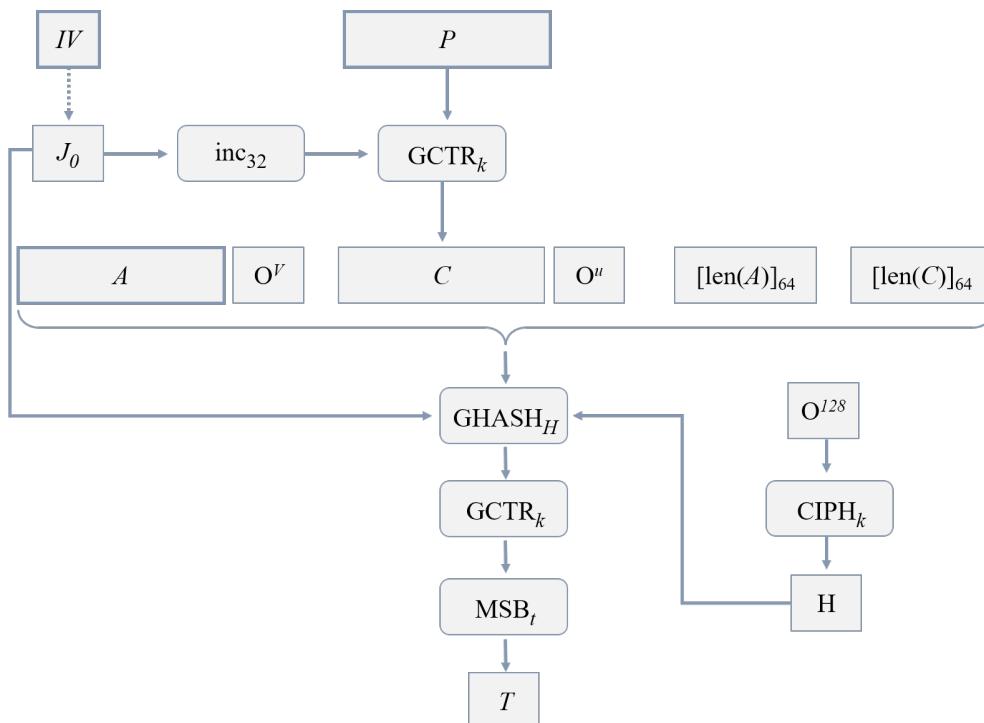
/* Η ετικέτα αυθεντικοποίησης (authentication tag) ορίζεται στο NIST 800-38D [28] ως ένας κρυπτογραφικός έλεγχος αθροίσματος σε δεδομένα που έχει σχεδιαστεί για να αποκαλύπτει τόσο τυχαία λάθη όσο και την εσκεμμένη τροποποίηση των δεδομένων. */

- 1 $H = CIPH(0^{128})$ // όπου $CIPH_k(X)$ η κρυπτογράφηση του μηνύματος X με το κλειδί k με τη χρήση του εγκεκριμένου αλγορίθμου $CIPH$.
 - 2 Ορίζουμε το μπλοκ J_0 ως ακολούθως:
 - 3 **if** $len(IV) = 96$ **then**
 - 4 | $J_0 = IV \parallel 0^{31} \parallel 1$
 - 5 **end**
 - 6 **if** $len(IV) \neq 96$ **then**
 - 7 | $s = 128 \cdot \lceil len(IV)/128 \rceil - len(IV)$ and $J_0 = GHASH_H(IV \parallel 0^{s+64} \parallel [len(IV)]_{64})$
 - 8 **end**
 - 9 $C = GCTR_K(inc_{32}(J_0), P)$
 - 10 $u = 128 \cdot \lceil len(C)/128 \rceil - len(C)$
 - 11 $v = 128 \cdot \lceil len(A)/128 \rceil - len(A)$
 - 12 $S = GHASH_H(A \parallel 0^v \parallel C \parallel 0^u \parallel [len(A)]_{64} \parallel [len(C)]_{64})$
 - 13 $T = MSB_t(GCTR_K(J_0, S))$ // Επέλεξε από το $GCTR_K(J_0, S)$ τα t bits από αριστερά.
 - 14 **return** (C, T)
-

Στο βήμα 1, δημιουργείται το κλειδί σύνοψης H για τη συνάρτηση GHASH, με την εφαρμογή του κρυπτογραφικού αλγορίθμου μπλοκ στο μπλοκ «μηδέν» (το μπλοκ που αποτελείται από ακολουθία 128 bits με τιμή “0”). Στα βήματα 2-8, το μπλοκ προ-μετρητή (pre-counter) J_0 δημιουργείται από το διάνυσμα αρχικοποίησης IV. Ειδικότερα, όταν το μήκος του IV είναι 96 bit, τότε η συμβολοσειρά πλήρωσης $0^{31} \parallel 1$ προσαρτάται στο IV για να σχηματίσει το μπλοκ προ-μετρητή. Διαφορετικά, το IV συμπληρώνεται με τον ελάχιστο αριθμό bits με τιμή “0”, έτσι ώστε το μήκος της συμβολοσειράς που προκύπτει να είναι πολλαπλάσιο των 128 bit (το μέγεθος του μπλοκ του κρυπτογραφικού αλγορίθμου). Αυτή η συμβολοσειρά με τη σειρά της επισυνάπτεται με 64 επιπλέον bits με τιμή “0”, ακολουθούμενα από την 64-bit αναπαράσταση του μήκους του IV, και η συνάρτηση GHASH εφαρμόζεται στη συμβολοσειρά που προκύπτει για να σχηματίσει το μπλοκ προ-μετρητή. Στο βήμα 9, η συνάρτηση αύξησης των 32 bits εφαρμόζεται στο μπλοκ προ-μετρητή για να παραχθεί το αρχικό μπλοκ μετρητή για μια κλήση της συνάρτησης GCTR (Galois/Counter Mode Counter) στο αρχικό κείμενο. Η έξοδος αυτής της κλήσης της συνάρτησης GCTR είναι το κρυπτογραφημένο κείμενο.

Στα βήματα 10 και 11-12, τα πρόσθετα αυθεντικοποιημένα δεδομένα A και το κρυπτογραφημένο κείμενο C προσαρτώνται το καθένα με τον ελάχιστο αριθμό από bits με τιμή “0”, έτσι ώστε τα μήκη των συμβολοσειρών που προκύπτουν να είναι πολλαπλάσια του προκαθορισμένου μεγέθους του μπλοκ του κρυπτογραφικού αλγορίθμου. Η συνένωση αυτών των συμβολοσειρών προσαρτάται με τις αναπαραστάσεις σε 64 bits των μηκών των πρόσθετων αυθεντικοποιημένων δεδομένων και του κρυπτογραφημένου κειμένου και η συνάρτηση

GHASH εφαρμόζεται στο αποτέλεσμα για να παραχθεί ένα ενιαίο μπλοκ εξόδου. Στο βήμα 13, αυτό το μπλοκ εξόδου κρυπτογραφείται χρησιμοποιώντας τη συνάρτηση GCTR με το μπλοκ προ-μετρητή που δημιουργήθηκε στα βήματα 2-8 και το αποτέλεσμα περικόπτεται στο καθορισμένο μήκος ετικέτας για να σχηματιστεί η ετικέτα αυθεντικοποίησης. Τέλος, το κρυπτογραφημένο κείμενο C και η ετικέτα T επιστρέφονται ως έξοδος.



Σχήμα 5.3: Αλγόριθμος αυθεντικοποιημένης κρυπτογράφησης GCM.

Οι Iwata και άλλοι [29] έδειξαν ότι οι αρχικοί ισχυρισμοί για την ασφάλεια του συστήματος GCM είναι εσφαλμένοι. Γι'αυτό πρότειναν μια διορθωμένη απόδειξη όπου υποθέτουν πως το IV είναι ένα nonce και ο υποκείμενος κρυπτογραφικός αλγόριθμος μπλοκ είναι ασφαλής. Επιπλέον, προκειμένου να αποτραπούν επιθέσεις που βασίζονται σε MAC μικρού μήκους, είναι σκόπιμο τα MAC να έχουν μήκος τουλάχιστον 96 bit. Επίσης το μήκος των nonces θα πρέπει να είναι 96 bit. Αδύναμα κλειδιά στο GCM έχουν εντοπιστεί από τους Handschuh και Preneel [30], Saarinen [31] και Procter και Cid [32].

Γενικώς το GCM θεωρείται ένα σχήμα που χρήζει ιδιαίτερης προσοχής καθώς λανθασμένες επιλογές παραμέτρων ή μικρά λάθη υλοποίησης μπορεί να έχουν σημαντικές συνέπειες. Κατά την ανάπτυξη του GCM, θα πρέπει να ελέγχονται προσεκτικά οι παράμετροι και οι λεπτομέρειες υλοποίησης. To NIST SP 800-38D [28] παρέχει σε παραρτήματά του σχετικές οδηγίες.

5.3.6.1 Συνάρτηση GHASH

Η συνάρτηση GHASH είναι ένα κρυπτογραφικό εργαλείο που χρησιμοποιείται για την επαλήθευση της ακεραιότητας και της αυθεντικότητας των δεδομένων. Συγκεκριμένα, χρησιμοποιείται στη λειτουργία GCM (Galois/Counter Mode) για την παραγωγή του κώδικα αυθεντικοποίησης μηνύματος (MAC).

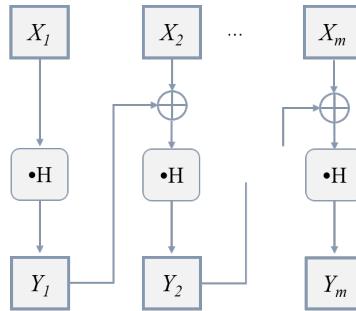
Η GHASH λαμβάνει ως είσοδο ένα μήνυμα m και ένα κλειδί k , που είναι αποτέλεσμα της κρυπτογράφησης του μπλοκ με τιμή 0 με το κύριο κλειδί κρυπτογράφησης. Το μήνυμα διαιρείται σε μπλοκ των 128 bits και κάθε μπλοκ επεξεργάζεται με τη χρήση της λειτουργίας XOR και της πολλαπλασιαστικής λειτουργίας σε ένα σώμα Galois $GF(2^{128})$. Η λειτουργία του αλγορίθμου περιγράφεται στον Αλγόριθμο 5.5 και απεικονίζεται στο Σχήμα 5.4.

Αλγόριθμος 5.5: Διαδικασία υπολογισμού της συνάρτησης $\text{GHASH}_H(X)$.

Είσοδος: Μια ακολουθία δεδομένων X με μήκος X_{len} ;

Έξοδος: Μια τιμή $\text{GHASH } Y_m$;

- 1 Έστω $X_1, X_2, \dots, X_{m-1}, X_m$ η μοναδική ακολουθία των μπλοκ τέτοια ώστε $X = X_1 \| X_2 \| \dots \| X_{m-1} \| X_m$
- 2 Έστω Y_0 το μπλοκ «μηδέν», δηλ. 0^{128}
- 3 **for** $1 \leq i \leq m$ **do**
- 4 $Y_i = (Y_{i-1} \oplus X_i) \bullet H$ // όπου $X \bullet Y$ το γινόμενο μεταξύ των μπλοκ X και Y σύμφωνα με το NIST 800-38D [28].
- 5 **end**
- 6 **return** Y_m



Σχήμα 5.4: Υπολογισμός της συνάρτησης $\text{GHASH}_H(X_1 \| X_2 \| \dots \| X_m) = Y_m$.

Αλγόριθμος 5.6: Διαδικασία υπολογισμού της συνάρτησης $\text{GCTR}_K(\text{ICB}, X)$.

Είσοδος: Εγκεκριμένος αλγόριθμος κρυπτογράφησης μπλοκ CIPH με μέγεθος μπλοκ 128-bit;
Κλειδί K ;
Αρχικό μπλοκ μετρητή ICB ;
Αρχικό κείμενο X αυθαίρετου μήκους;

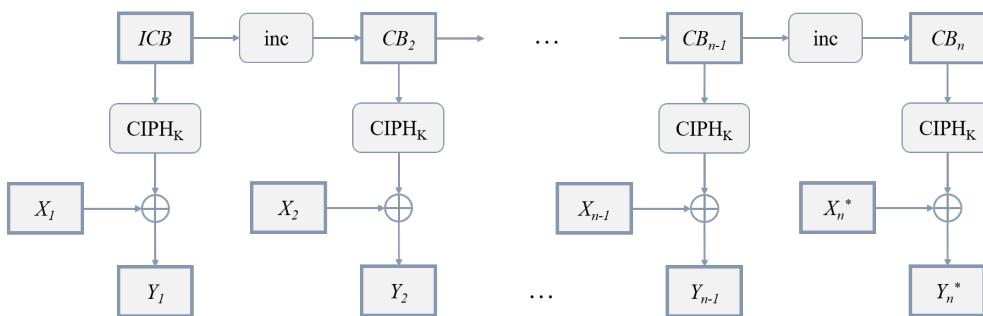
Έξοδος: Κρυπτογραφημένο κείμενο Y με μήκος bit $\text{len}(X)$;

- 1 Αν το X είναι η κενή συμβολοσειρά, τότε επιστρέψτε την κενή συμβολοσειρά ως Y .
- 2 $n = \left\lceil \frac{\text{len}(X)}{128} \right\rceil$
- 3 Έστω $X_1, X_2, \dots, X_{n-1}, X_n^*$ η μοναδική ακολουθία από bit strings τέτοια ώστε
 $X = X_1 \| X_2 \| \dots \| X_{n-1} \| X_n^*$; όπου X_1, X_2, \dots, X_{n-1} είναι πλήρη blocks.
- 4 $CB_1 = ICB$
- 5 **for** $2 \leq i \leq n$ **do**
- 6 $CB_i = \text{inc32}(CB_{i-1})$
- 7 **end**
- 8 **for** $1 \leq i \leq n-1$ **do**
- 9 $Y_i = X_i \oplus \text{CIPH}_K(CB_i)$
- 10 **end**
- 11 $Y_n^* = X_n^* \oplus \text{MSB}_{\text{len}(X_n^*)}(\text{CIPH}_K(CB_n))$
- 12 $Y = Y_1 \| Y_2 \| \dots \| Y_n^*$
- 13 **return** Y

5.3.6.2 Συνάρτηση GCTR

Η συνάρτηση GCTR (βλέπε Αλγόριθμο 5.6) χρησιμοποιείται για την κρυπτογράφηση και αποκρυπτογράφηση δεδομένων σε λειτουργία GCM. Χρησιμοποιεί έναν μετρητή (counter) που αυξάνεται για κάθε μπλοκ δεδομένων που επεξεργάζεται, και κρυπτογραφεί αυτά τα μπλοκ χρησιμοποιώντας το κύριο κλειδί κρυπτογράφησης.

Στα βήματα 1 και 2, η είσοδος, αυθαίρετου μήκους, χωρίζεται σε μια ακολουθία από ακέραια μπλοκ στο μεγαλύτερο δυνατό βαθμό, έτσι ώστε μόνο το δεξιότερο μπλοκ στην ακολουθία να μπορεί να είναι ένα “μερικό” μπλοκ. Στα βήματα 3 και 4, η συνάρτηση αύξησης των 32 bit επαναλαμβάνεται στο αρχικό μπλοκ μετρητή για να δημιουργηθεί μια ακολουθία από μπλοκ μετρητών. Το μπλοκ εισόδου είναι το πρώτο μπλοκ της ακολουθίας. Στα βήματα 5-7 και 8-10, η συμμετρική κρυπτογράφηση εφαρμόζεται στα μπλοκ μετρητών και τα αποτελέσματα χρησιμοποιούνται ως είσοδος σε μια συνάρτηση XOR με τα αντίστοιχα μπλοκ (ή μερικά μπλοκ) της διαίρεσης της εισόδου. Στα βήματα 11-12, η ακολουθία των αποτελεσμάτων συνενώνεται για να σχηματίσει την έξοδο, όπως απεικονίζεται στο Σχήμα 5.5.



Σχήμα 5.5: Υπολογισμός της συνάρτησης $GCTR_K(ICB, X_1||X_2||\dots||X_n^*) = Y_1||Y_2||\dots||Y_n^*$.

5.3.7 ChaCha20+Poly1305

Το ChaCha20+Poly1305 προκύπτει από το συνδυασμό κρυπτογράφησης με τον αλγόριθμο ροής ChaCha20 και υπολογισμού σύνοψης με τη συνάρτηση Poly1305 χρησιμοποιώντας τον συνδυασμό Κρυπτογράφηση-και-μετά-MAC, όπως ακριβώς η λειτουργία GCM, με τη διαφορά ότι το κλειδί αλλάζει με κάθε κρυπτογράφηση. Περιγράφεται στο RFC 7905 [33] και η ασφάλεια του αναλύεται στο [34]. Επειδή το κλειδί ενημερώνεται με κάθε κρυπτογράφηση, καταφέρνει να αποφύγει πολλές πιθανές ευπάθειες που υπάρχουν στο GCM.

5.4 Ψηφιακές Υπογραφές

Στην ενότητα αυτή παρουσιάζονται προτυποποιημένα και μη σχήματα ψηφιακών υπογραφών τα οποία μπορούν να χρησιμοποιηθούν για την ακεραιότητα μηνύματος και την αυθεντικοποίηση αποστολέα. Στον Πίνακα 5.4 παρουσιάζονται τα σχήματα ψηφιακών υπογραφών που συνιστώνται, και μη, για συστήματα παλαιού τύπου καθώς και για μελλοντική χρήση.

Η δημιουργία και επαλήθευση υπογραφής γίνεται σύμφωνα με τρεις βασικούς τύπους σχημάτων:

- Ψηφιακή υπογραφή με ανάκτηση μηνύματος (Signature with message recovery)
- Ψηφιακή υπογραφή με μερική ανάκτηση μηνύματος (Signature with partial message recovery)
- Ψηφιακή υπογραφή με προσθήκη (Signature with appendix)

Και στους τρεις τύπους σχημάτων το σύστημα ψηφιακής υπογραφής αποτελείται από:

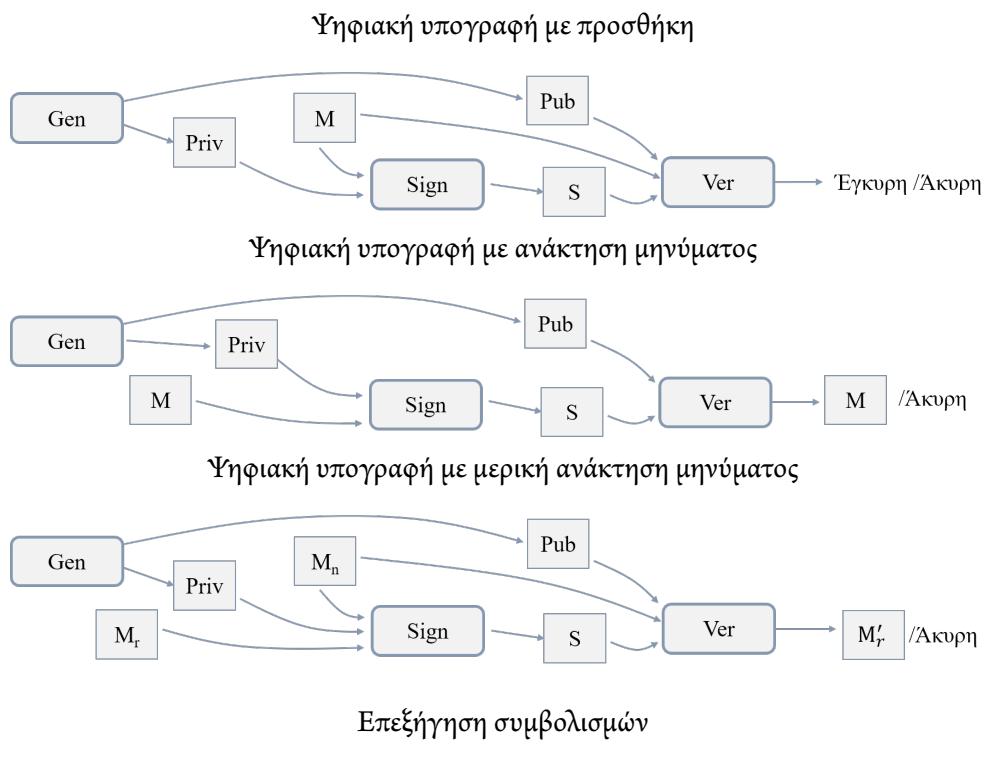
Πίνακας 5.4: Συστήματα ψηφιακών υπογραφών βασισμένα σε κρυπτογραφία δημοσίου κλειδιού.

Σχήμα	Κατηγοριοποίηση		Σημειώσεις
	Παλαιού Τύπου	Μελλοντική Χρήση	
RSA-PSS	✓	✓	
ISO-9796-2 RSA-DS2	✓	✓	Παραλλαγή του RSA-PSS που παρέχει ανάκτηση μηνύματος
RSA-PKCS #1 v1.5	✓	✓	
PV Signatures	✓	✓	
(EC) Schnorr	✓	✓	
(EC) KDSA	✓	✓	
EdDSA	✓	✓	
XMSS	✓	✓	
RSA-FDH	✓	✗	Ζητήματα κατά την προετοιμασία της απαιτούμενης συνάρτησης σύνοψης
ISO-9796-2 RSA-DS3	✓	✗	
RSA-FDH	✓	✗	Παρόμοια με το RSA-FDH
(EC) DSA, (EC) GDSA	✓	✗	Τα επίπεδα ασφαλείας δεν είναι επαρκώς αποδεδειγμένα
(EC) RDSA	✓	✗	Τα επίπεδα ασφαλείας δεν είναι επαρκώς αποδεδειγμένα
ISO-9796-2 RSA-DS1	✗	✗	

- μια λειτουργία δημιουργίας υπογραφής, η οποία παράγει μια υπογραφή από ένα μήνυμα με τη χρήση του ιδιωτικού κλειδιού του υπογράφοντος, και
- μια λειτουργία επαλήθευσης υπογραφής, με την οποία επαληθεύεται η υπογραφή στο μήνυμα με το αντίστοιχο δημόσιο κλειδί του υπογράφοντος.

Η διαφορά μεταξύ των σχημάτων που παρέχουν ανάκτηση, έστω και μερική, του μηνύματος, με το σχήμα της ψηφιακής υπογραφής με προσθήκης, είναι πως στην περίπτωση του τελευταίου, που είναι και το πιο συνηθισμένο σχήμα υπογραφών, η υπογραφή εφαρμόζεται στη σύνοψη του μηνύματος και όχι στο ίδιο το μήνυμα, όπως γίνεται στην ψηφιακή υπογραφή με ανάκτηση μηνύματος. Έτσι, ένα σχήμα ψηφιακής υπογραφής με προσθήκη μπορεί να χρησιμοποιηθεί για την υπογραφή μηνυμάτων οποιουδήποτε μήκους, επειδή η σύνοψη του μηνύματος, η οποία είναι συνήθως 256-512 bits, δεν υπερβαίνει το μέγιστο μέγεθος του μηνύματος που προκύπτει από το modulus του αλγορίθμου. Η υπογραφή της σύνοψης του μηνύματος είναι ασφαλής εφόσον η συνάρτηση σύνοψης είναι ανθεκτική σε συγκρούσεις. Στην περίπτωση των ψηφιακών υπογραφών με (μερική) ανάκτηση μηνύματος, τμήμα του αρχικού κειμένου ή ολόκληρο το κείμενο ανακτάται κατά τη διαδικασία της επαλήθευσης της υπογραφής.

Για να επαληθεύουμε μια υπογραφή που έχει δημιουργηθεί με ψηφιακή υπογραφή με προσθήκη, είναι απαραίτητο να έχουμε το ίδιο το (αρχικό) μήνυμα καθώς η υπογραφή δημιουργείται στη σύνοψη του μηνύματος. Στην περίπτωση της ψηφιακής υπογραφής με ανάκτηση μηνύματος, το μήνυμα, ή τμήμα αυτού, ανακτάται κατά τη διαδικασία της επαλήθευσης και επομένως δεν απαιτείται η πρότερη γνώση του μηνύματος για την επαλήθευση της υπογραφής. Ο τρόπος λειτουργίας των τριών σχημάτων απεικονίζεται στο Σχήμα 5.6:



Σχήμα 5.6: Βασικά σχήματα υπογραφών.

5.4.1 RSA-PKCS #1

Το RSA-PKCS #1 v1.5 [35] είναι ένα σχήμα υπογραφής με προσθήκη, με ευρεία αποδοχή. Αποτελεί μέρος της οικογένειας των Public Key Cryptography Standards (PKCS) τα οποία αναπτύχθηκαν από την RSA Laboratories σε συνεργασία με μια άτυπη κοινοπραξία, που αρχικά περιελάμβανε τις Apple, Microsoft, DEC, Lotus, Sun και MIT. Χρησιμοποιείται σε έναν μεγάλο αριθμό σημαντικών προτύπων και εφαρμογών, όπως τα X.509 (RFC4055 [36]), S/MIME (RFC3370 [37]), PGP (RFC4880 [38]), IPSec (RFC4359 [39]), όλες οι εκδόσεις του Transport Layer Security (TLS) πρωτοκόλλου (βλέπε Ενότητα 10.2) έως και την έκδοση 1.2 (στην έκδοση 1.3 γίνεται χρήση του σχήματος RSA-PSS (βλέπε Ενότητα 10.2)), JSON WebS XMLSignature του W3C και πολλές άλλες.

Η διαδικασία της ψηφιακής υπογραφής είναι η τυπική που ακολουθείται για τον αλγόριθμο RSA και περιλαμβάνει τα ακόλουθα στάδια και βήματα:

- Δημιουργία ζεύγους κλειδιών:
 - Ο υπογράφων δημιουργεί ένα ζεύγος κλειδιών που αποτελείται από ένα ιδιωτικό κλειδί και ένα αντίστοιχο δημόσιο κλειδί σύμφωνα με τις απαιτήσεις του αλγορίθμου.
- Δημιουργία υπογραφής:

- Υπολογίζεται η σύνοψη του μηνύματος M χρησιμοποιώντας έναν ασφαλή αλγόριθμο κατακερματισμού (π.χ. SHA-256):

$$H = \text{Hash}(M)$$

- Δημιουργείται το *DigestInfo*, το οποίο περιλαμβάνει τη συνάρτηση σύνοψης και την σύνοψη H :

$$\text{DigestInfo} = T \parallel H$$

όπου T είναι το ASN.1 DER encoding της συνάρτησης σύνοψης.

- Συμπληρώνεται το *DigestInfo* σε ένα block δεδομένων σύμφωνα με το σχήμα κρυπτογράφησης RSASSA-PKCS1-v1_5 [35]:

$$EM = 0x00 \parallel 0x01 \parallel PS \parallel 0x00 \parallel \text{DigestInfo}$$

όπου PS είναι η ακολουθία των bytes πλήρωσης με τιμή 0xFF και μήκος τέτοιο ώστε να συμπληρωθεί το συνολικό μήκος του μπλοκ.

- Κρυπτογραφείται το μπλοκ των δεδομένων EM με το ιδιωτικό κλειδί d για να δημιουργηθεί η ψηφιακή υπογραφή:

$$S = EM^d \bmod n$$

- Επαλήθευση υπογραφής

- Αποκρυπτογραφείται η ψηφιακή υπογραφή με τη χρήση του δημοσίου κλειδιού του υπογράφοντος, λαμβάνοντας έτσι την συμπληρωμένη τιμή σύνοψης.
- Αφαιρείται η πλήρωση από την ανακτηθείσα σύνοψη και υπολογίζεται εκ νέου η σύνοψη του ληφθέντος μηνύματος χρησιμοποιώντας την ίδια κρυπτογραφική συνάρτηση σύνοψης.
- Συγκρίνεται η υπολογισμένη τιμή σύνοψης με την ανακτηθείσα τιμή σύνοψης. Εάν είναι ίδιες, η υπογραφή θεωρείται έγκυρη. Διαφορετικά, είναι άκυρη.

5.4.2 RSA-PSS

To RSA PKCS #1 v2.1 εισήγαγε το RSA Probabilistic Signature Scheme (RSA-PSS), ένα βελτιωμένο σχήμα πιθανοτικών υπογραφών με προσθήκη, το οποίο σχεδιάστηκε για να κάνει την υπογραφή μηνυμάτων πιο ασφαλή, χάρη στην προσθήκη τυχαίων δεδομένων πλήρωσης. Αυτό σημαίνει ότι στη διαδικασία δημιουργίας της υπογραφής εισάγεται και μια τυχαία τιμή που δημιουργείται από μια γεννήτρια ψευδοτυχαίων αριθμών του υπογράφοντα. Ο παραλήπτης του μηνύματος μπορεί στη συνέχεια να επαληθεύσει την υπογραφή χρησιμοποιώντας το αντίστοιχο δημόσιο κλειδί RSA. Με τη χρήση των τυχαίων δεδομένων πλήρωσης δύο υπογραφές με την ίδια είσοδο καταλήγουν να είναι διαφορετικές. Ωστόσο μπορούν να χρησιμοποιηθούν και οι δύο για την επικύρωση των αρχικών δεδομένων.

Η διαδικασία δημιουργίας μιας ψηφιακής υπογραφής στο μήνυμα m με τη χρήση του σχήματος RSA-PSS περιλαμβάνει τα εξής βήματα [35]:

- Υπολογισμός της σύνοψης του μηνύματος m χρησιμοποιώντας μια ασφαλή συνάρτηση σύνοψης (π.χ. SHA-256):

$$H = \text{Hash}(m)$$

- Δημιουργία μιας τυχαίας τιμής $salt$ μήκους $sLen$ bytes.

- Δημιουργία του μηνύματος M' :

$$M' = \text{Padding1} \parallel H \parallel \text{salt}$$

όπου το Padding1 είναι μια ακολουθία μηδενικών bytes για την επίτευξη του επιθυμητού μήκους, ώστε το μήκος του M' να είναι:

$$\text{length}(M') = \text{emLen} - \text{sLen} - \text{hLen} - 2$$

όπου emLen (encoded message length) είναι το μήκος του κωδικοποιημένου μηνύματος σε bytes², και hLen το μήκος της σύνοψης σε bytes.

- Εφαρμογή της συνάρτησης MGF1 (Mask Generation Function) στη σύνοψη H για τη δημιουργία του dbMask (η MGF1 αναλύεται στην Ενότητα 5.4.2.1):

$$\text{dbMask} = \text{MGF1}(H, k - \text{hLen} - 1)$$

όπου k το μήκος του κλειδιού RSA σε bytes.

- Δημιουργία του maskedDB με τη χρήση της συνάρτησης XOR:

$$\text{maskedDB} = (0x00 \parallel \text{PS} \parallel 0x01 \parallel \text{salt}) \oplus \text{dbMask}$$

- Δημιουργία του κωδικοποιημένου μηνύματος (encoded message) EM :

$$EM = \text{maskedDB} \parallel H \parallel 0xbc$$

- Κρυπτογράφηση του EM με το ιδιωτικό κλειδί (d, n) για να δημιουργηθεί η ψηφιακή υπογραφή (το RSASP1 αναλύεται στην ενότητα 5.4.2.2):

$$S = \text{RSASP1}(d, n, EM)$$

Η διαδικασία επαλήθευσης μιας ψηφιακής υπογραφής S με το σχήμα RSA-PSS στο μήνυμα M , περιλαμβάνει τα εξής βήματα:

- Μετατροπή της υπογραφής S στον ακέραιο s χρησιμοποιώντας την συνάρτηση OS2IP (Octet String to Integer Primitive) [35], η οποία μετατρέπει μια ακολουθία bytes σε έναν μη αρνητικό ακέραιο αριθμό:

$$s = \text{OS2IP}(S)$$

- Υπολογισμός του κωδικοποιημένου μηνύματος EM με τη χρήση της συνάρτησης RSAVP1 και του δημοσίου κλειδιού (e, n) :

$$EM = \text{RSAVP1}(e, n, s)$$

όπου RSAVP1 (RSA Verification Primitive) είναι η αντίστροφη λειτουργία του RSASP1 (βλέπε Ενότητα 5.4.2.2).

- Το κωδικοποιημένο μήνυμα EM μετατρέπεται σε μια ακολουθία bytes χρησιμοποιώντας την συνάρτηση I2OSP (Integer to Octet String Primitive) και χωρίζεται στα επιμέρους τμήματα:

$$EM = \text{maskedDB} \parallel H' \parallel 0xbc$$

όπου H' η σύνοψη του μηνύματος που βρίσκεται μέσα στο κωδικοποιημένο μήνυμα EM .

²To emLen πρέπει να είναι ίσο με το μήκος του modulus n σε bytes. Δηλαδή, αν το modulus n είναι 2048 bits (256 bytes), τότε το emLen είναι 256 bytes.

- Δημιουργία της μάσκας δεδομένων $dbMask$ από το H' με τη χρήση της συνάρτησης MGF1:

$$dbMask = \text{MGF1}(H', k - hLen - 1)$$

- Υπολογισμός της ακολουθίας δεδομένων DB χρησιμοποιώντας το $dbMask$ με της χρήση της συνάρτησης XOR:

$$DB = \text{maskedDB} \oplus dbMask$$

- Έλεγχος της εγκυρότητας της δομής του DB :

$$DB = \text{Padding2} \parallel salt \parallel 0x01 \parallel H'$$

- Υπολογισμός της σύνοψης H του μηνύματος m :

$$H = \text{Hash}(m)$$

- Έλεγχος αν η σύνοψη H είναι ίση με την H' . Αν είναι, τότε η υπογραφή θεωρείται έγκυρη:

$$H \stackrel{?}{=} H'$$

5.4.2.1 Mask Generation Function

Η MGF είναι μια κρυπτογραφική συνάρτηση που χρησιμοποιείται για την παραγωγή μιας ακολουθίας ψευδοτυχαίων bytes από μια δεδομένη είσοδο σταθερού μήκους. Στο πλαίσιο του RSA-PSS, η MGF1 χρησιμοποιείται για τη δημιουργία μιας μάσκας που εφαρμόζεται με τη χρήση της συνάρτησης XOR σε ένα μπλοκ δεδομένων. Συγκεκριμένα, η MGF1 παίρνει ως είσοδο τη σύνοψη του μηνύματος και δημιουργεί μια μάσκα που χρησιμοποιείται για την παραγωγή του maskedDB , ως εξής:

- Το επιθυμητό μήκος μάσκας $maskLen$ διαιρείται σε πολλαπλάσια του μήκους της συνάρτησης σύνοψης $hLen$.
- Για κάθε τιμή σύνοψης $Counter$, υπολογίζεται:

$$T = \text{Hash}(\text{Seed} \parallel Counter)$$

όπου $Seed$ είναι η αρχική είσοδος και $Counter$ μια αύξουσα τιμή από 0 μέχρι $\lceil maskLen/hLen \rceil - 1$.

- Όλες οι τιμές T συνενώνονται για να δημιουργηθεί η μάσκα:

$$\text{mask} = T_0 \parallel T_1 \parallel \dots \parallel T_{\lceil maskLen/hLen \rceil - 1}$$

5.4.2.2 RSASP1

Το RSASP1 είναι μια βασική λειτουργία υπογραφής με τον αλγόριθμο RSA, και η οποία περιλαμβάνει τα ακόλουθα βήματα για την κρυπτογράφηση του κωδικοποιημένου μηνύματος EM με τη χρήση του ιδιωτικού κλειδιού.

- Το κωδικοποιημένο μήνυμα EM , το οποίο είναι μια ακολουθία bytes αντιμετωπίζεται ως ένας ακέραιος m με τη μορφή:

$$m = \text{OS2IP}(EM)$$

όπου OS2IP (Octet String to Integer Primitive) μετατρέπει την ακολουθία bytes σε ακέραιο.

- Υπολογίζεται η υπογραφή s χρησιμοποιώντας το ιδιωτικό κλειδί d και το δημόσιο modulus n :

$$s = m^d \bmod n$$

- Η υπογραφή s μετατρέπεται σε ακολουθία bytes S με τη μορφή:

$$S = \text{I2OSP}(s, k)$$

όπου I2OSP (Integer to Octet String Primitive) συνάρτηση που μετατρέπει τον ακέραιο σε ακολουθία bytes μήκους k .

5.4.3 (EC)DSA

Ο αλγόριθμος ψηφιακής υπογραφής (Digital Signature Algorithm – DSA) (Ενότητα 2.5) και η παραλλαγή του με κρυπτογραφία ελλειπτικών καμπύλων (ECDSA) (βλέπε Ενότητα 2.6.3) είναι ευρέως τυποποιημένοι αλγόριθμοι [40, 41] και αποτελούν, μαζί με τον RSA, τους αλγορίθμους που είναι εγκεκριμένοι από το NIST Digital Signature Standard (DSS) [40]. Επιπλέον, υπάρχει ένας σημαντικός αριθμός παραλλαγών του DSA που έχουν υιοθετηθεί από διάφορες άλλες χώρες, συμπεριλαμβανομένου του γερμανικού DSA (GDSA), του κορεατικού DSA (KDSA) και του ρωσικού DSA (RDSA).

Τα επίπεδα ασφαλείας όλων των παραλλαγών του (EC)DSA (εκτός του KDSA) δεν έχουν αποδειχτεί επαρκώς. Ο λόγος είναι ότι η συνάρτηση σύνοψης εφαρμόζεται μόνο στο μήνυμα και όχι στον συνδυασμό του μηνύματος και του εφήμερου κλειδιού, όπως γίνεται με τις PV υπογραφές (βλέπε Ενότητα 5.4.4). Ο αλγόριθμος KDSA ανήκει στην κατηγορία των κατάλληλων για μελλοντική χρήση αλγορίθμων.

5.4.4 PV Signatures

Το ISO 14888-3 [42] όρισε μια παραλλαγή των υπογραφών DSA, όπου η διαδικασία της υπογραφής είναι ακριβώς ίδια με τον DSA, με εξαίρεση τη σύνοψη η οποία υπολογίζεται στη συνένωση του μηνύματος και ενός εφήμερου κλειδιού, το οποίο αποτελεί μια τυχαία τιμή η οποία χρησιμοποιείται για την υπογραφή ενός συγκεκριμένου μηνύματος μόνο. Αυτό το σχήμα υπογραφών προτάθηκε από τους Pointcheval και Vaudenay [43], και ως εκ τούτου, συχνά αναφέρεται ως σχήμα υπογραφής PV.

Η διαδικασία ψηφιακής υπογραφής περιλαμβάνει τα ακόλουθα βήματα:

- Δημιουργία κλειδιών:
 - Δημιουργία Ιδιωτικού Κλειδιού (Private Key Generation): Επιλέγεται ένας τυχαίος ακέραιος x από ένα μεγάλο πεδίο \mathbb{Z}_q , όπου q είναι ένας μεγάλος πρώτος αριθμός.
 - Δημιουργία Δημοσίου Κλειδιού (Public Key Generation): Υπολογίζεται το δημόσιο κλειδί y ως $y = g^x \text{ mod } p$, όπου g είναι μια γεννήτρια της κυκλικής ομάδας \mathbb{Z}_p^* , και p είναι ένας μεγάλος πρώτος αριθμός.
- Δημιουργία υπογραφής ένα μήνυμα m :
 - Επιλέγεται ένας τυχαίος ακέραιος k , ο οποίος αποτελεί και το εφήμερο κλειδί, από το πεδίο \mathbb{Z}_q .
 - Υπολογίζεται το $r = g^k \text{ mod } p$.
 - Υπολογίζεται η σύνοψη του μηνύματος μαζί με το r , δηλαδή $e = H(m||r)$, όπου H είναι συνάρτηση σύνοψης.
 - Υπολογίζεται η υπογραφή s ως $s = k - x \cdot e \text{ mod } q$.

Η τελική υπογραφή αποτελείται από το ζεύγος (r, s) .

- Επαλήθευση της υπογραφής (r, s) για το μήνυμα m :
 - Υπολογίζεται η τιμή $e = H(m||r)$.
 - Υπολογίζονται οι τιμές $v_1 = g^s \text{ mod } p$ και $v_2 = y^e \text{ mod } p$.
 - Ελέγχεται αν $r = v_1 \cdot v_2 \text{ mod } p$.

Αν η παραπάνω εξίσωση ισχύει, τότε η υπογραφή είναι έγκυρη.

Ομοίως με τις υπογραφές (EC)DSA, οι υπογραφές PV αντιμετωπίζουν ζητήματα που σχετίζονται με κακή τυχαιότητα στο εφήμερο κλειδί. Έχουν πολλά από τα πλεονεκτήματα των υπογραφών Schnorr. Ωστόσο, οι υπογραφές Schnorr έχουν μια πιο απλή στην εφαρμογή διαδικασία υπογραφής. Ενώ έχουν καθοριστεί μόνο για πεπερασμένα πεδία στο ISO 14888-3, μπορούν να επεκταθούν και στις ελλειπτικές καμπύλες.

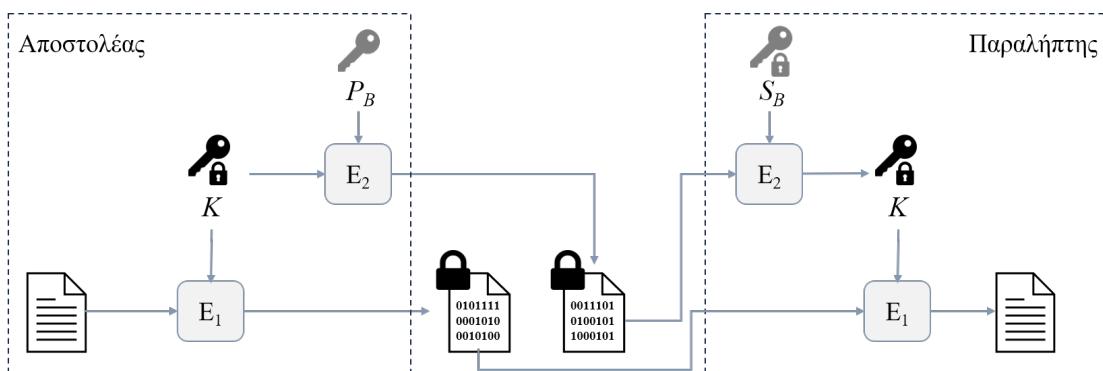
5.5 Ασκήσεις-Εργασίες

Εργασίες

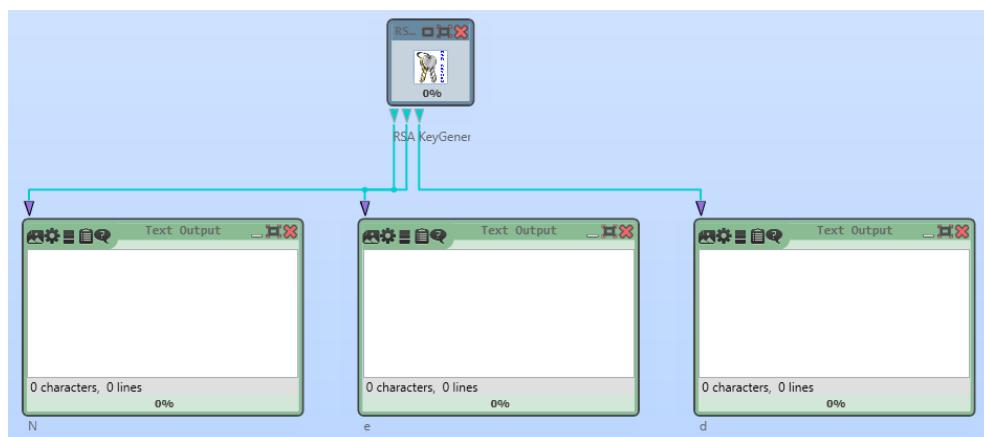
5.5.1 Χρησιμοποιώντας την εφαρμογή [CrypTool 2](#), δημιουργήστε τη συνάρτηση HMAC της Ενότητας 5.2.2.

5.5.2 Χρησιμοποιώντας την εφαρμογή [CrypTool 2](#), και το κρυπτοσύστημα του υποερωτήματος (2) της Εργασίας 1.6.1 ως βάση, τροποποιήστε το κατάλληλα ώστε ο αποστολέας του μηνύματος να μπορεί να δώσει στον παραλήπτη το κλειδί της συμμετρικής κρυπτογράφησης, κρυπτογραφώντας το με το δημόσιο κλειδί του παραλήπτη, εξασφαλίζοντας έτσι την εμπιστευτικότητα του κλειδιού (σύμφωνα με το Σχήμα 5.7). Για τη δημιουργία των ασύμμετρων κλειδών, θα χρησιμοποιήσετε το component του Cryptool2 με όνομα RSA Key Generator (Σχήμα 5.8).

Ο μηχανισμός αυτός αποτελεί τη βάση για τη μετάδοση κρυπτογραφημένων emails, σύμφωνα με το πρωτόκολλο SMIME [44].



Σχήμα 5.7: Κρυπτογράφηση συμμετρικού κλειδιού με τη χρήση κρυπτογραφίας δημοσίου κλειδιού.



Σχήμα 5.8: Γεννήτρια κλειδιών RSA.

5.5.3 Χρησιμοποιώντας την εφαρμογή [CrypTool 2](#), και το κρυπτοσύστημα της Εργασίας 5.5.2 να κάνετε όλες τις απαραίτητες προθήκες/αλλαγές ώστε επιπλέον της κρυπτογράφησης να υπογράφετε και το μη κρυπτογραφημένο αρχείο και να επαληθεύετε την υπογραφή σε αυτό. Για την επαλήθευση της υπογραφής χρησιμοποιήστε και το component “Comparators”.

Προφανώς για την υπογραφή θα πρέπει να χρησιμοποιήσετε ένα δεύτερο ζεύγος κλειδιών, αυτό του αποστολέα.

5.5.4 Χρησιμοποιώντας την εφαρμογή [CrypTool 2](#), και το κρυπτοσύστημα του υποερωτήματος (2) της Εργασίας 1.6.1 ως βάση, τροποποιήστε το κατάλληλα ώστε να επιτύχετε αυθεντικοποιημένη κρυπτογράφηση με την μέθοδο *Κρυπτογράφηση-και-μετά-MAC* (Σχήμα 5.2) για αρχείο εισόδου της επιλογής σας. Η κρυπτογράφηση θα πρέπει να γίνει με τη χρήση του αλγορίθμου AES σε τρόπο λειτουργίας CBC και κλειδί κρυπτογράφησης μεγέθους 128 bits, ενώ για MAC μπορείτε να χρησιμοποιήσετε το HMAC με συνάρτηση.

Βιβλιογραφία

- [1] Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 2018.
- [2] M J Dworkin. *Recommendation for block cipher modes of operation :: the CMAC mode for authentication*. en. Tech. rep. NIST SP 800-38b. Edition: 0. Gaithersburg, MD: National Institute of Standards and Technology, 2016, NIST SP 800-38b. doi: [10.6028/NIST.SP.800-38b](https://doi.org/10.6028/NIST.SP.800-38b).
- [3] ISO/IEC 9797-1:2011. *Information technology — Security techniques — Digital signatures giving message recovery — Part 1: Mechanisms using a block cipher*. 2011. URL: www.iso.org.
- [4] John Kelsey, Shu-jen Chang, and Ray Perlner. *SHA-3 derived functions: cSHAKE, KMAC, TupleHash and ParallelHash*. Tech. rep. NIST SP 800-185. Gaithersburg, MD: National Institute of Standards and Technology, Dec. 2016, NIST SP 800-185. doi: [10.6028/NIST.SP.800-185](https://doi.org/10.6028/NIST.SP.800-185).
- [5] Erez Petrank and Charles Rackoff. “CBC MAC for Real-Time Data Sources”. en. In: *Journal of Cryptology* 13.3 (June 2000), pp. 315–338. ISSN: 0933-2790, 1432-1378. doi: [10.1007/s001450010009](https://doi.org/10.1007/s001450010009).
- [6] Krzysztof Pietrzak. “A Tight Bound for EMAC”. In: *Automata, Languages and Programming*. Ed. by David Hutchison et al. Vol. 4052. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 168–179. ISBN: 978-3-540-35907-4 978-3-540-35908-1. doi: [10.1007/11787006_15](https://doi.org/10.1007/11787006_15).
- [7] ANSI X9.19: *Financial Institution Retail Message Authentication*. ANSI X9.19-1996. American National Standards Institute, 1996.
- [8] Tetsu Iwata and Kaoru Kurosawa. “OMAC: One-Key CBC MAC”. In: *Fast Software Encryption*. Ed. by Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, and Thomas Johansson. Vol. 2887. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 129–153. ISBN: 978-3-540-20449-7 978-3-540-39887-5. doi: [10.1007/978-3-540-39887-5_11](https://doi.org/10.1007/978-3-540-39887-5_11).
- [9] Mridul Nandi. “A Unified Method for Improving PRF Bounds for a Class of Blockcipher Based MACs”. In: *Fast Software Encryption*. Ed. by David Hutchison et al. Vol. 6147. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 212–229. ISBN: 978-3-642-13857-7 978-3-642-13858-4. doi: [10.1007/978-3-642-13858-4_12](https://doi.org/10.1007/978-3-642-13858-4_12).
- [10] ISO/IEC 9797-2:2011. *Information technology — Security techniques — Digital signatures giving message recovery — Part 2: Mechanisms using a dedicated hash-function*. 2011. URL: www.iso.org.
- [11] H. Krawczyk, M. Bellare, and R. Canetti. *HMAC: Keyed-Hashing for Message Authentication*. en. Tech. rep. RFC2104. RFC Editor, Feb. 1997, RFC2104. doi: [10.17487/rfc2104](https://doi.org/10.17487/rfc2104).
- [12] National Institute of Standards and Technology. *The Keyed-Hash Message Authentication Code (HMAC)*. Tech. rep. NIST FIPS 198-1. Gaithersburg, MD: National Institute of Standards and Technology, July 2008, NIST FIPS 198-1. doi: [10.6028/NIST.FIPS.198-1](https://doi.org/10.6028/NIST.FIPS.198-1).

- [13] John Kelsey and Bruce Schneier. “Second preimages on n-bit hash functions for much less than 2n work”. In: *Annual International Workshop on Selected Areas in Cryptography*. Springer. 2005, pp. 474–490.
- [14] Mihir Bellare and Chanathip Namprempre. *Authenticated Encryption: Relations among notions and analysis of the generic composition paradigm*. Cryptology ePrint Archive, Paper 2000/025. 2000. URL: <https://eprint.iacr.org/2000/025>.
- [15] N. J. Al Fardan and K. G. Paterson. “Lucky Thirteen: Breaking the TLS and DTLS Record Protocols”. In: *2013 IEEE Symposium on Security and Privacy*. Berkeley, CA: IEEE, May 2013, pp. 526–540. ISBN: 978-0-7695-4977-4 978-1-4673-6166-8. DOI: 10.1109/SP.2013.42.
- [16] ISO/IEC 19972:2009. *Information technology — Security techniques — Digital signatures giving message recovery — Authenticated encryption*. 2009. URL: www.iso.org.
- [17] Phillip Rogaway, Mihir Bellare, and John Black. “OCB: A block-cipher mode of operation for efficient authenticated encryption”. en. In: *ACM Transactions on Information and System Security* 6.3 (Aug. 2003), pp. 365–403. ISSN: 1094-9224, 1557-7406. DOI: 10.1145/937527.937529.
- [18] Daniel Bleichenbacher. “Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1”. In: *Annual International Cryptology Conference*. Springer. 1998, pp. 1–12.
- [19] Phillip Rogaway. “Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC”. In: *Advances in Cryptology - ASIACRYPT 2004*. Ed. by David Hutchison et al. Vol. 3329. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 16–31. ISBN: 978-3-540-23975-8 978-3-540-30539-2. DOI: 10.1007/978-3-540-30539-2_2.
- [20] Akiko Inoue, Tetsu Iwata, Kazuhiko Minematsu, and Bertram Poettering. “Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality”. en. In: *Advances in Cryptology – CRYPTO 2019*. Ed. by Alexandra Boldyreva and Daniele Micciancio. Vol. 11692. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2019, pp. 3–31. ISBN: 978-3-030-26947-0 978-3-030-26948-7. DOI: 10.1007/978-3-030-26948-7_1.
- [21] MJ Dworkin. *Recommendation for block cipher modes of operation :: the CCM mode for authentication and confidentiality*. en. Tech. rep. NIST SP 800-38c. Edition: 0. Gaithersburg, MD: National Institute of Standards and Technology, 2007, NIST SP 800–38c. DOI: 10.6028/NIST.SP.800-38c.
- [22] Jakob Jonsson. “On the Security of CTR + CBC-MAC”. en. In: *Selected Areas in Cryptography*. Ed. by Kaisa Nyberg and Howard Heys. Vol. 2595. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 76–93. ISBN: 978-3-540-00622-0 978-3-540-36492-4. DOI: 10.1007/3-540-36492-7_7.
- [23] P. Rogaway and D. Wagner. *A Critique of CCM*. Cryptology ePrint Archive, Paper 2003/070. 2003. URL: <https://eprint.iacr.org/2003/070>.
- [24] Mihir Bellare, Phillip Rogaway, and David Wagner. “The EAX Mode of Operation”. In: *Fast Software Encryption*. Ed. by Takeo Kanade et al. Vol. 3017. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 389–407. ISBN: 978-3-540-22171-5 978-3-540-25937-4. DOI: 10.1007/978-3-540-25937-4_25.

- [25] Tadayoshi Kohno, John Viega, and Doug Whiting. “CWC: A High-Performance Conventional Authenticated Encryption Mode”. In: *Fast Software Encryption*. Ed. by Takeo Kanade et al. Vol. 3017. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 408–426. ISBN: 978-3-540-22171-5 978-3-540-25937-4. DOI: 10.1007/978-3-540-25937-4_26.
- [26] D. McGrew. “Efficient Authentication of Large, Dynamic Data Sets Using Galois/Counter Mode (GCM)”. In: *Third IEEE International Security in Storage Workshop (SISW'05)*. San Francisco, CA, USA: IEEE, 2005, pp. 89–94. ISBN: 978-0-7695-2537-2. DOI: 10.1109/SISW.2005.3.
- [27] David A. McGrew and John Viega. “The Security and Performance of the Galois/Counter Mode (GCM) of Operation”. In: *Progress in Cryptology - INDOCRYPT 2004*. Ed. by David Hutchison et al. Vol. 3348. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 343–355. ISBN: 978-3-540-24130-0 978-3-540-30556-9. DOI: 10.1007/978-3-540-30556-9_27.
- [28] M J Dworkin. *Recommendation for block cipher modes of operation :: GaloisCounter Mode (GCM) and GMAC*. en. Tech. rep. NIST SP 800-38d. Edition: 0. Gaithersburg, MD: National Institute of Standards and Technology, 2007, NIST SP 800-38d. DOI: 10.6028/NIST.SP.800-38d.
- [29] Tetsu Iwata, Keisuke Ohashi, and Kazuhiko Minematsu. “Breaking and Repairing GCM Security Proofs”. In: *Advances in Cryptology – CRYPTO 2012*. Ed. by Reihaneh Safavi-Naini and Ran Canetti. Vol. 7417. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 31–49. ISBN: 978-3-642-32008-8 978-3-642-32009-5. DOI: 10.1007/978-3-642-32009-5_3.
- [30] Helena Handschuh and Bart Preneel. “Key-Recovery Attacks on Universal Hash Function Based MAC Algorithms”. en. In: *Advances in Cryptology – CRYPTO 2008*. Ed. by David Wagner. Vol. 5157. ISSN: 0302-9743, 1611-3349 Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 144–161. ISBN: 978-3-540-85173-8 978-3-540-85174-5. DOI: 10.1007/978-3-540-85174-5_9.
- [31] Markku-Juhani Olavi Saarinen. “Cycling Attacks on GCM, GHASH and Other Polynomial MACs and Hashes”. In: *Fast Software Encryption*. Ed. by Anne Canteaut. Vol. 7549. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 216–225. ISBN: 978-3-642-34046-8 978-3-642-34047-5. DOI: 10.1007/978-3-642-34047-5_13.
- [32] Gordon Procter and Carlos Cid. “On Weak Keys and Forgery Attacks Against Polynomial-Based MAC Schemes”. en. In: *Journal of Cryptology* 28.4 (Oct. 2015), pp. 769–795. ISSN: 0933-2790, 1432-1378. DOI: 10.1007/s00145-014-9178-9.
- [33] A. Langley, W. Chang, N. Mavrogiannopoulos, J. Strombergson, and S. Josefsson. *ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS)*. en. Tech. rep. RFC7905. RFC Editor, June 2016, RFC7905. DOI: 10.17487/RFC7905.
- [34] Gordon Procter. “A Security Analysis of the Composition of ChaCha20 and Poly1305”. In: *IACR Cryptol. ePrint Arch.* 2014 (2014), p. 613.
- [35] K. Moriarty, B. Kaliski, J. Jonsson, and A. Rusch. *PKCS #1: RSA Cryptography Specifications Version 2.2*. en. Tech. rep. RFC8017. RFC Editor, Nov. 2016, RFC8017. DOI: 10.17487/RFC8017.
- [36] J. Schaad, B. Kaliski, and R. Housley. *Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. en. Tech. rep. RFC4055. RFC Editor, June 2005, RFC4055. DOI: 10.17487/rfc4055.
- [37] R. Housley. *Cryptographic Message Syntax (CMS) Algorithms*. en. Tech. rep. RFC3370. RFC Editor, Aug. 2002, RFC3370. DOI: 10.17487/rfc3370.

- [38] J. Callas, L. Donnerhacke, H. Finney, D. Shaw, and R. Thayer. *OpenPGP Message Format*. en. Tech. rep. RFC4880. RFC Editor, Nov. 2007, RFC4880. doi: 10.17487/rfc4880.
- [39] B. Weis. *The Use of RSA/SHA-1 Signatures within Encapsulating Security Payload (ESP) and Authentication Header (AH)*. en. Tech. rep. RFC4359. RFC Editor, Jan. 2006, RFC4359. doi: 10.17487/rfc4359.
- [40] Dustin Moody. *Digital Signature Standard (DSS)*. Tech. rep. NIST FIPS 186-5. Gaithersburg, MD: National Institute of Standards and Technology, 2023, NIST FIPS 186-5. doi: 10.6028/NIST.FIPS.186-5.
- [41] ANSI X9.62, *Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*. American National Standards Institute, X9-Financial Services. 2020.
- [42] *IT Security techniques – Digital signatures with appendix – Part 3: Discrete logarithm based mechanisms*. Standard. Geneva, CH: International Organization for Standardization, Nov. 2018.
- [43] David Pointcheval and Serge Vaudenay. “On Provable Security for Digital Signature Algorithms”. In: (Nov. 1996).
- [44] Jim Schaad, Blake C. Ramsdell, and Sean Turner. *Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification*. RFC 8551. Apr. 2019. doi: 10.17487/RFC8551. URL: <https://www.rfc-editor.org/info/rfc8551>.

ΚΕΦΑΛΑΙΟ 6

ΔΙΑΧΕΙΡΙΣΗ ΚΡΥΠΤΟΓΡΑΦΙΚΩΝ ΚΛΕΙΔΙΩΝ

Περίληψη

Η κρυπτογραφία αποτελεί τη βάση των μηχανισμών προστασίας των πληροφοριών. Ωστόσο, η πλημμελής χρήση της ενέχει κινδύνους καθώς μπορεί να επηρεάσει σημαντικά την ασφάλεια ενός κρυπτοσυστήματος και κατ'επέκταση των περιουσιακών στοιχείων του οργανισμού που αυτό προστατεύει. Λαμβάνοντας υπόψη μια από τις βασικές αρχές στην κρυπτογραφία, αυτήν του Kerckhoffs, η ασφαλής διαχείριση των κρυπτογραφικών κλειδιών αποτελεί ένα σημαντικό θέμα στην κρυπτογραφία. Συνηθισμένες απειλές για ένα κρυπτοσύστημα αποτελούν η χρήση αριθμών ανεπαρκούς τυχαιότητας κατά τη δημιουργία των κλειδιών, η κακή ή λανθασμένη χρήση των κλειδιών και η ανασφαλής αποθήκευσή τους. Για τους παραπάνω, καθώς και για άλλους, λόγους, απαιτείται η υιοθέτηση πρόσθετων μηχανισμών, πολιτικών και διαδικασιών που αφορούν στην ασφαλή διαχείριση των κλειδιών, είτε πρόκειται για κλειδιά μακροχρόνιας χρήσης, είτε για εφήμερα κλειδιά ή κλειδιά συνεδρίας. Σε αυτή την ενότητα αναλύονται θέματα που σχετίζονται με τον κύκλο ζωής των κρυπτογραφικών κλειδιών (Ενότητα 6.1), τις απαιτήσεις προστασίας τους (Ενότητα 6.2) και την χρονική διάρκεια χρήσης τους (Ενότητα 6.3). Έμφαση δίνεται στη διαχείριση συμμετρικών κλειδιών (Ενότητα 6.4) όπου και αναλύονται πρωτόκολλα εδραίωσης κλειδιών (Ενότητα 6.5), όπως τα Needham-Schroeder, Diffie-Hellman, και το σύστημα Kerberos. Τέλος, αναλύονται θέματα που αφορούν στα σχήματα κοινής χρήσης μυστικών (Ενότητα 6.6), ενώ αναφορά γίνεται και στις μονάδες ασφαλείας υλισμικού (Ενότητα 6.7).

Προαπαιτούμενη γνώση: Κατανόηση των βασικών εννοιών της Κρυπτογραφίας που παρατίθενται στα εισαγωγικά κεφάλαια (Κεφάλαιο 1 έως 3) αυτού του βιβλίου.

6.1 Διαχείριση Κλειδιών

Η κρυπτογραφία είναι ο θεμέλιος λίθος για πολλούς μηχανισμούς ασφάλειας πληροφοριών. Ωστόσο, η μη ορθή χρήση της εισάγει σημαντικούς κινδύνους που ενδέχεται να επηρεάσουν την ασφάλεια των δεδομένων, με έμφαση στην εμπιστευτικότητα και την ακεραιότητά τους.

Οι ασφαλείς κρυπτογραφικοί μηχανισμοί περιορίζουν το πρόβλημα της ασφάλειας των δεδομένων σε αυτό

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

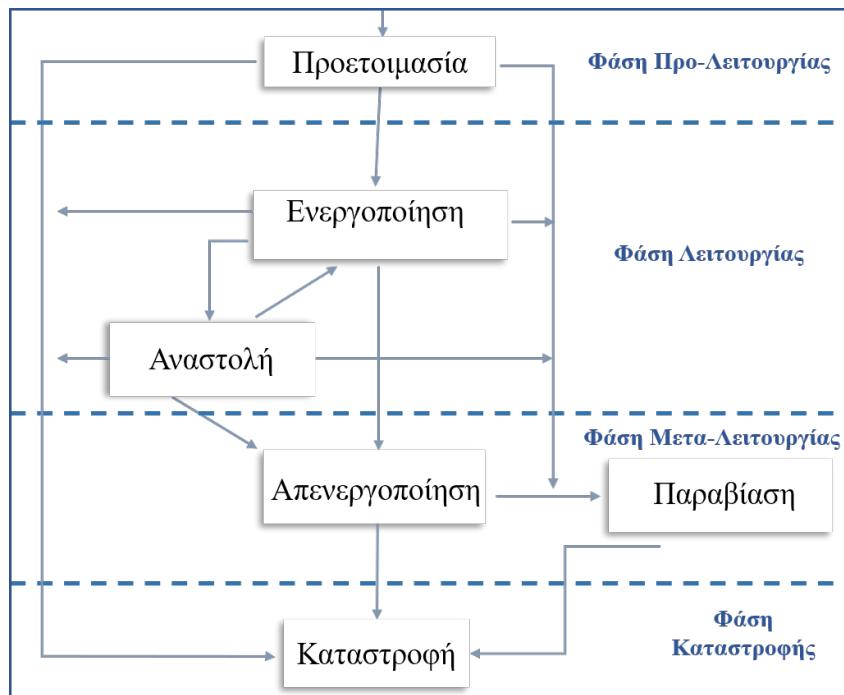
 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

της ασφαλούς διαχείρισης κρυπτογραφικών κλειδιών. Η αρχή του Kerckhoffs, σύμφωνα με την οποία «η ασφάλεια ενός κρυπτοσυστήματος δεν πρέπει να βασίζεται στη μυστικότητα οποιασδήποτε λειτουργίας του, εκτός από το μυστικό κρυπτογραφικό κλειδί», υποδηλώνει τη σημαντικότητα της προστασίας των κρυπτογραφικών κλειδιών από μη εξουσιοδοτημένη πρόσβαση. Αυτή η απειλή είναι ένας από τους λόγους για τους οποίους η διαχείριση των κρυπτογραφικών κλειδιών είναι ένα τόσο σημαντικό ζήτημα στην κρυπτογραφία.

Η διαχείριση κλειδιών αφορά ολόκληρο τον κύκλο ζωής τους – από τη δημιουργία τους μέχρι την καταστροφή τους. Οι βασικοί στόχοι της ορθής διαχείρισης κρυπτογραφικών κλειδιών συνοψίζονται στους ακόλουθους:

- Προστασία της εμπιστευτικότητας και της ακεραιότητας των μυστικών και ιδιωτικών κλειδιών.
- Προστασία των μυστικών και ιδιωτικών κλειδιών από μη εξουσιοδοτημένη χρήση.
- Προστασία της ορθότητας των δημόσιων κλειδιών και της συσχέτισής τους με τον εξουσιοδοτημένο κάτοχο τους.
- Διασφάλιση της διαθεσιμότητας των κλειδιών.
- Διασφάλιση της ορθής και ασφαλούς χρήσης, κατάλληλων για το εκάστοτε σύστημα, κρυπτογραφικών κλειδιών.

Ο κύκλος ζωής ενός κρυπτογραφικού κλειδιού, σύμφωνα με το NIST 800-57 [1, 2], αποτελείται από τέσσερις κύριες φάσεις (Σχήμα 6.1): προ-λειτουργίας, λειτουργίας, μετα-λειτουργίας και καταστροφής.



Σχήμα 6.1: Κύκλος ζωής κρυπτογραφικών κλειδιών κατά NIST (NIST 800-57 [1]).

6.1.1 Φάση Προ-Λειτουργίας

Η φάση προ-λειτουργίας περιλαμβάνει όλα τα απαραίτητα βήματα για την προετοιμασία του περιβάλλοντος για την ασφαλή χρήση κρυπτογραφικών κλειδιών. Αυτά τα βήματα απαιτούν την εγγραφή των εξουσιοδοτημένων οντοτήτων σε έναν τομέα του συστήματος διαχείρισης κλειδιών και την εγκατάσταση των απαραίτητων

αρχικών κλειδιών εντός του λογισμικού, του υλικού, του συστήματος, της εφαρμογής, της κρυπτογραφικής μονάδας ή της συσκευής. Μπορούν επίσης να απαιτήσουν τον καθορισμό κλειδιών μεταξύ των συμμετεχουσών οντοτήτων, χρησιμοποιώντας είτε πρωτόκολλα μεταφοράς κλειδιών (Ενότητα 6.4.1.2), είτε πρωτόκολλα συμφωνίας κλειδιών (Ενότητα 6.5.2). Σε αυτή τη φάση τα κρυπτογραφικά κλειδιά δεν είναι ακόμα διαθέσιμα για χρήση.

Εάν το σύστημα υλοποιεί ένα κρυπτοσύστημα δημοσίου κλειδιού, πρέπει να δημιουργηθούν ισχυρά ζεύγη κλειδιών σύμφωνα με τις απαιτήσεις των αλγορίθμων. Τα δημόσια κλειδιά πρέπει να πιστοποιούνται από έναν αξιόπιστο πάροχο υπηρεσιών εμπιστοσύνης (Κεφάλαιο 7) και να γίνονται διαθέσιμα σε όλες τις συμμετέχουσες οντότητες που θέλουν να τα χρησιμοποιήσουν. Εάν είναι απαραίτητο, τα κλειδιά μπορούν να δημιουργούνται σε ελεγχόμενο περιβάλλον, όπως μια μονάδα ασφαλείας υλικού (Ενότητα 6.7) ή σε περιβάλλον ανθεκτικό σε παραβιάσεις, χρησιμοποιώντας ασφαλείς γεννήτριες τυχαίων αριθμών και σύμφωνα με τις απαιτήσεις των αλγορίθμων για τους οποίους δημιουργούνται.

6.1.2 Φάση Λειτουργίας

Κατά τη διάρκεια της φάσης λειτουργίας, τα κλειδιά χρησιμοποιούνται για τους σκοπούς που έχουν δημιουργηθεί και παραμένουν λειτουργικά μέχρι το τέλος της κρυπτοπεριόδου τους (Ενότητα 6.3).

Η φάση λειτουργίας περιλαμβάνει τους μηχανισμούς και τις διαδικασίες που θα διασφαλίσουν ότι το κλειδί θα παραμείνει λειτουργικό και προσβάσιμο σε εξουσιοδοτημένες οντότητες, ακόμη και αν χαθεί ή δεν μπορεί να χρησιμοποιηθεί λόγω βλάβης υλικού, φθοράς ή απώλειας μέσων αποθήκευσης τους. Οι μηχανισμοί που μπορούν να αναπτυχθούν για το σκοπό αυτό υλοποιούν λειτουργίες δημιουργίας αντιγράφων ασφαλείας και ανάκτησης. Η ανάκτηση κλειδιού (key recovery) περιλαμβάνει μηχανισμούς και διαδικασίες που επιτρέπουν σε εξουσιοδοτημένες οντότητες να ανακτούν ή να ανακατασκευάζουν κλειδιά ή/και άλλες πληροφορίες σχετικές με τα κλειδιά, από αντίγραφα ασφαλείας ή άλλα βασικά στοιχεία.

Οι διαδικασίες αλλαγής κλειδιών αποτελούν επίσης αντικείμενο της φάσης λειτουργίας, ενώ είναι ιδιαίτερα σημαντικές στις περιπτώσεις όπου χρειάζεται να αντικατασταθούν κλειδιά που έχουν παραβιαστεί ή υπάρχουν υποψίες ότι έχουν παραβιαστεί. Σε μια τέτοια περίπτωση το κλειδί μπαίνει σε φάση αναστολής (key suspension) όπως φαίνεται και στο Σχήμα 6.1 προκειμένου να διερευνηθεί η δυνατότητα επαναχρησιμοποίησής του και, ως αποτέλεσμα, είτε να αντικατασταθεί, είτε να ενεργοποιηθεί και πάλι.

Τα κλειδιά θα πρέπει να αντικαθίστανται όταν η κρυπτοπερίοδός τους πλησιάζει στη λήξη, αλλά και για τον περιορισμό του όγκου των δεδομένων που προστατεύονται από το ίδιο κλειδί. Κατά τη διάρκεια της φάσης λειτουργίας, μπορούν επίσης να χρησιμοποιηθούν μηχανισμοί για την δημιουργία κλειδιών συνεδρίας ή συνόδου ή εφήμερων κλειδιών από άλλα βασικά δεδομένα. Η διαδικασία αυτή είναι γνωστή και ως παραγγή κλειδιού (key derivation) και μπορεί να είναι αποτέλεσμα εκτέλεσης κάποιου πρωτοκόλλου εδραίωσης κλειδιών (key establishment protocol) (Ενότητα 6.5).

6.1.3 Φάση Μετα-Λειτουργίας

Ένα κρυπτογραφικό κλειδί μπαίνει στη φάση μετα-λειτουργίας όταν δεν χρησιμοποιείται ή δεν πρέπει να χρησιμοποιείται πλέον, επειδή έχει λήξει η κρυπτοπερίοδός του, έχει παραβιαστεί ή έχει αντικατασταθεί. Σε αυτήν τη φάση, το κλειδί μπορεί να εξακολουθεί να είναι προσβάσιμο για ορισμένους λόγους, όπως για να εξασφαλιστεί η πρόσβαση σε δεδομένα που έχουν κρυπτογραφηθεί με αυτό το κλειδί. Επομένως, προκειμένου να ανακτηθεί το κλειδί σε μεταγενέστερο στάδιο, πρέπει να αρχειοθετηθεί και να διατηρηθεί για μια συγκεκριμένη περίοδο.

Η αρχειοθέτηση κλειδιών δεν απαιτείται για όλα τα κρυπτογραφικά κλειδιά. Στην πραγματικότητα υπάρχουν κλειδιά που δεν πρέπει να αρχειοθετούνται επειδή δεν χρειάζεται να χρησιμοποιηθούν μετά τη λήξη της κρυπτοπεριόδου τους. Για παράδειγμα, τα ιδιωτικά κλειδιά υπογραφής ή τα ιδιωτικά κλειδιά που χρησιμοποιούνται για σκοπούς ελέγχου ταυτότητας, δεν θα χρησιμοποιηθούν ξανά μετά τις ημερομηνίες λήξης τους.

Κατά τη φάση μετα-λειτουργίας πραγματοποιείται επίσης η αποσύνδεσή του από τον κάτοχό του και η δια-

γραφή του κατόχου από το σύστημα. Αυτό σημαίνει πως το κλειδί δεν σχετίζεται πλέον με ένα συγκεκριμένο σύστημα ή τομέα.

6.1.4 Φάση Καταστροφής

Κατά τη διάρκεια της φάσης καταστροφής τα κλειδιά δεν είναι πλέον διαθέσιμα σε καταχωρημένες οντότητες και πρέπει να διαγραφούν με ασφάλεια από οποιαδήποτε συσκευή αποθήκευσης.

6.2 Απαιτήσεις Προστασίας Κρυπτογραφικών Κλειδιών

Τα κρυπτογραφικά κλειδιά, όπως αναφέρθηκε παραπάνω, θα πρέπει να προστατεύονται σε όλη τη διάρκεια του κύκλου ζωής τους. Βασικές απαιτήσεις για την προστασία τους αποτελούν αδιαμφισβήτητα η εμπιστευτικότητα, η ακεραιότητα και η διαθεσιμότητά τους. Οι απαιτήσεις αυτές μπορούν να ικανοποιηθούν με τη χρήση μιας κρυπτογραφικής μονάδας, εσωτερικά στον οργανισμό, ή με τη βοήθεια έμπιστων τρίτων οντοτήτων και των υπηρεσιών προστασίας που αυτές παρέχουν. Μια πιο λεπτομερής αναφορά στις επιμέρους απαιτήσεις προστασίας κρυπτογραφικών κλειδιών περιλαμβάνει τα εξής:

- **Εμπιστευτικότητα:** Θα πρέπει να διασφαλίζεται για όλες τις πληροφορίες που αφορούν ένα μυστικό κλειδί ή σχετίζονται άμεσα με αυτό, όπως: συμμετρικά κλειδιά, ιδιωτικά κλειδιά, και κοινά στοιχεία κλειδιών. Τα δημόσια κλειδιά και πολλά από τα μεταδεδομένα των κλειδιών, γενικά δεν απαιτούν προστασία εμπιστευτικότητας. Όταν οι πληροφορίες μυστικού κλειδιού φυλάσσονται σε μια κρυπτογραφική μονάδα, η οποία καλύπτει τις απαιτήσεις διεθνών προτύπων όπως είναι το ISO/IEC 19790:2012 [3] και το FIPS [4], παρέχεται κατάλληλη προστασία εμπιστευτικότητας. Όταν οι πληροφορίες μυστικού κλειδιού είναι διαθέσιμες εκτός κρυπτογραφικής μονάδας, η προστασία του απορρήτου παρέχεται με κρυπτογράφηση σε κατάλληλο επίπεδο ασφαλείας ή με έλεγχο φυσικής πρόσβασης. Η ασφάλεια και ο λειτουργικός αντίκτυπος συγκεκριμένων μηχανισμών προστασίας της εμπιστευτικότητας ποικίλει.
- **Ακεραιότητα:** Θα πρέπει να διασφαλίζεται για όλες τις βασικές πληροφορίες και περιλαμβάνουν, μεταξύ άλλων, τυχαίους αριθμούς που αποτελούν τη βάση για τη δημιουργία κλειδιών, κρυπτογραφικά κλειδιά και σχετικά μεταδεδομένα (π.χ. κάτοχος του κλειδιού, χρόνος ζωής). Η προστασία ακεραιότητας απαιτεί πάντα τον έλεγχο της πηγής και της μορφής των βασικών πληροφοριών που λαμβάνονται ή ανακτώνται. Όπως και με την εμπιστευτικότητα, οι κρυπτογραφικές μονάδες μπορούν να παρέχουν και προστασία της ακεραιότητας των κρυπτογραφικών κλειδιών.. Όταν τα κλειδιά είναι διαθέσιμα εκτός κρυπτογραφικής μονάδας, η προστασία ακεραιότητας θα πρέπει να παρέχεται από κατάλληλους μηχανισμούς ακεραιότητας (π.χ. MAC ή ψηφιακές υπογραφές), ή μηχανισμούς φυσικής προστασίας.
- **Προστασία συσχέτισης:** Θα πρέπει να παρέχεται για κάθε κρυπτοσύστημα, διασφαλίζοντας ότι το σωστό κρυπτογραφικό υλικό χρησιμοποιείται για την προστασία των σωστών δεδομένων στη σωστή εφαρμογή ή εξοπλισμό.
- **Διασφάλιση κατοχής ιδιωτικού κλειδιού:** παρέχει διαβεβαιώσεις ότι ο κάτοχος ενός δημόσιου κλειδιού κατέχει πράγματι το αντίστοιχο ιδιωτικό κλειδί.
- **Διαθεσιμότητα:** Θα πρέπει να διασφαλίζεται για όλες τις βασικές πληροφορίες του κλειδιού που πρέπει να είναι διαθέσιμες πέρα από την άμεση χρήση τους για την προστασία δεδομένων (π.χ. για την αποκρυπτογράφηση ακόμα και μετά το πέρας ισχύος του κλειδιού).

Η περίοδος προστασίας των βασικών πληροφοριών εξαρτάται από τον τύπο του κλειδιού, το κρυπτοσύστημα και τη σχετική υπηρεσία χρήσης αυτών, καθώς και το χρονικό διάστημα για το οποίο χρησιμοποιείται το κρυπτοσύστημα, και περιλαμβάνει την κρυπτοπερίοδο του κλειδιού. Δεν είναι απαραίτητα ίδια για την ακεραιότητα και την εμπιστευτικότητα. Για παράδειγμα, προστασία ακεραιότητας μπορεί να απαιτείται έως ότου

ένα κλειδί δεν χρησιμοποιείται πλέον (αλλά δεν έχει καταστραφεί ακόμη), αλλά προστασία εμπιστευτικότητας μπορεί να απαιτείται έως ότου καταστραφεί πραγματικά το κλειδί.

6.3 Κρυπτοπερίοδος

Η κρυπτοπερίοδος (cryptoperiod) είναι το χρονικό διάστημα κατά το οποίο ένα συγκεκριμένο κλειδί εξουσιοδοτείται για χρήση από νόμιμες οντότητες ή κατά το οποίο τα κλειδιά για ένα δεδομένο σύστημα παραμένουν σε ισχύ. Μια κατάλληλα καθορισμένη κρυπτοπερίοδος συμβάλλει ουσιαστικά στην ασφάλεια του κρυπτοσυστήματος καθώς:

1. Περιορίζει τον όγκο των πληροφοριών που είναι διαθέσιμες για κρυπτανάλυση, π.χ. το πλήθος ζευγών αρχικού και κρυπτογραφημένου κειμένου που είναι κρυπτογραφημένα με το κλειδί.
2. Περιορίζει το μέγεθος της έκθεσης των δεδομένων που προστατεύονται με ένα κλειδί σε περίπτωση παραβίασης αυτού.
3. Περιορίζει τον διαθέσιμο χρόνο για προσπάθειες διείσδυσης σε φυσικούς, διαδικαστικούς και λογικούς μηχανισμούς πρόσβασης που προστατεύονται ένα κλειδί από μη εξουσιοδοτημένη αποκάλυψη.
4. Περιορίζει την περίοδο εντός της οποίας οι πληροφορίες ενδέχεται να τεθούν σε κίνδυνο λόγω ακούσιας αποκάλυψης ενός κρυπτογραφικού κλειδιού σε μη εξουσιοδοτημένες οντότητες.
5. Περιορίζει τον διαθέσιμο χρόνο για υπολογιστικά εντατική κρυπτανάλυση.

6.3.1 Παράμετροι που Επηρεάζουν τη Διάρκεια

Μεταξύ των παραγόντων που επηρεάζουν τη διάρκεια μιας κρυπτοπεριόδου είναι:

1. Τα χαρακτηριστικά του κρυπτοσυστήματος, όπως ο αλγόριθμος, το μήκος του κλειδιού και ο τρόπος λειτουργίας.
2. Η ενσωμάτωση των μηχανισμών, π.χ. με υλοποίηση σε κρυπτογραφική μονάδα ή μέσω εφαρμογής λογισμικού σε προσωπικό υπολογιστή.
3. Το περιβάλλον λειτουργίας, π.χ. χρήση μιας ασφαλούς εγκατάστασης περιορισμένης πρόσβασης, ή παροχή πρόσβασης από ένα κοινό τερματικό.
4. Εναλλαγή προσωπικού, π.χ. διαχειριστών συστημάτων και προσωπικού ενός Παρόχου Υπηρεσιών Εμπιστοσύνης (Κεφάλαιο 7).
5. Ο όγκος της ροής δεδομένων ή το πλήθος των συναλλαγών που προστατεύονται με το συγκεκριμένο κλειδί.
6. Το χρονικό διάστημα για το οποίο απαιτείται προστασία των δεδομένων.
7. Η προβλεπόμενη χρήση των κλειδιών, π.χ. κρυπτογράφηση δεδομένων, ψηφιακή υπογραφή, παραγωγή κλειδιού ή προστασία κλειδιού.
8. Η μέθοδος αντικατάστασης κλειδιών, π.χ. εισαγωγή μέσω πληκτρολογίου, χρήση μονάδας ασφαλείας υλισμικού (Ενότητα 6.7) όπου οι χρήστες δεν έχουν άμεση πρόσβαση στα κλειδιά ή απομακρυσμένη αντικατάσταση σε μια υποδομή δημοσίου κλειδιού.
9. Το πλήθος των οντοτήτων που μοιράζονται ένα κοινό κλειδί.

10. Το πλήθος των αντιγράφων ενός κλειδιού και ο τρόπος διαμοιρασμού αυτών των αντιγράφων.
11. Τα κίνητρα των επιτιθέμενων για τις προστατευμένες πληροφορίες.
12. Η απειλή για τις προστατευμένες πληροφορίες από νέες και ανατρεπτικές τεχνολογίες, π.χ. κβαντικοί υπολογιστές.

Γενικά, οι σύντομες κρυπτοπερίοδοι ενισχύουν την ασφάλεια. Για παράδειγμα, ορισμένοι αλγόριθμοι κρυπτογράφησης μπορεί να είναι λιγότερο ευάλωτοι στην κρυπτανάλυση εάν ο επιτιθέμενος έχει πρόσβαση σε περιορισμένο αριθμό πληροφοριών κρυπτογραφημένες με το ίδιο κλειδί. Από την άλλη πλευρά, σε περιπτώσεις όπου οι μέθοδοι μη αυτοματοποιημένου διαμοιρασμού κλειδιών υπόκεινται σε ανθρώπινο λάθος και αδυναμία, οι πιο συχνές αλλαγές κλειδιών ενδέχεται στην πραγματικότητα να αυξήσουν τον κίνδυνο έκθεσης των κλειδιών αυτών. Σε αυτές τις περιπτώσεις, μπορεί να είναι πιο συνετό να έχουμε σπανιότερες, καλά ελεγχόμενες, αυτοματοποιημένους διαμοιρασμούς κλειδιών παρά πιο συχνές, κακώς ελεγχόμενες μη αυτοματοποιημένους διαμοιρασμούς κλειδιών.

Οι συνέπειες της αποκάλυψης ενός κλειδιού επίσης επηρεάζουν την κρυπτοπερίοδο και τυπικά σχετίζονται με την ευαισθησία των πληροφοριών, την κρισιμότητα των διαδικασιών που προστατεύονται από το κρυπτοσύστημα και το κόστος ανάκτησης ως αποτέλεσμα της παραβίασης των διαδικασιών που προστατεύονται από τη χρήση κρυπτογραφίας. Η ευαισθησία αφορά στη διάρκεια ζωής των πληροφοριών που προστατεύονται (π.χ. 10 λεπτά, 10 ημέρες ή 10 χρόνια) και τις πιθανές συνέπειες από την απώλεια της προστασίας αυτών των πληροφοριών (π.χ. αποκάλυψη των πληροφοριών σε μη εξουσιοδοτημένες οντότητες).

Γενικά, καθώς αυξάνεται η ευαισθησία των πληροφοριών ή η κρισιμότητα των διαδικασιών που προστατεύονται από την κρυπτογραφία, η διάρκεια των συσχετισμένων κρυπτοπεριόδων θα πρέπει να μειώνεται προκειμένου να περιοριστεί η ζημιά που μπορεί να προκύψει από κάθε αποκάλυψη των κλειδιών και των προστατευμένων με αυτά, δεδομένων. Ωστόσο, οι σύντομες κρυπτοπερίοδοι μπορεί να είναι αντιπαραγωγικές.

Άλλοι παράγοντες που μπορεί να επηρεάσουν την κρυπτοπερίοδο αφορούν στην κατάσταση των δεδομένων που κρυπτογραφούνται και το κόστος της ανάκτησης και αντικατάστασης των κλειδιών.

- Τα κλειδιά που χρησιμοποιούνται για την προστασία των δεδομένων σε κατάσταση μεταφοράς συχνά έχουν μικρότερες κρυπτοπεριόδους από τα κλειδιά που χρησιμοποιούνται για την προστασία των αποθηκευμένων δεδομένων (σε κατάσταση ηρεμίας). Οι κρυπτοπερίοδοι γενικά γίνονται μεγαλύτερες για αποθηκευμένα δεδομένα, επειδή η επιβάρυνση της εκ νέου κρυπτογράφησης όλων των δεδομένων που κρυπτογραφήθηκαν χρησιμοποιώντας τα παλιά κλειδιά μπορεί να είναι κοστοβόρα.
- Σε ορισμένες περιπτώσεις, το κόστος που σχετίζεται με την αλλαγή/αντικατάσταση κλειδιών είναι υψηλό. Παραδείγματα περιλαμβάνουν την αποκρυπτογράφηση και την επακόλουθη επανακρυπτογράφηση πολύ μεγάλων βάσεων δεδομένων, την αποκρυπτογράφηση και την εκ νέου κρυπτογράφηση κατανευμένων βάσεων δεδομένων και την ανάκληση και αντικατάσταση ενός πολύ μεγάλου αριθμού κλειδιών (π.χ. όπου υπάρχει πολύ μεγάλος αριθμός κατόχων κλειδιών γεωγραφικά και οργανωτικά κατανευμένων). Σε τέτοιες περιπτώσεις, μπορεί να δικαιολογηθεί η επιπλέον δαπάνη των μέτρων ασφαλείας που απαιτούνται για την υποστήριξη μακροχρόνιων κρυπτοπεριόδων (π.χ. η χρήση κρυπτογραφίας που μπορεί να οδηγήσει σε επιπλέον έξοδα επεξεργασίας).

6.3.2 Κρυπτοπερίοδοι Συμμετρικών Κλειδιών

Η χρονική περίοδος κατά την οποία χρησιμοποιείται ένα συμμετρικό κλειδί για την κρυπτογράφηση δεδομένων ή τη δημιουργία ενός MAC διαφέρει από τη χρονική περίοδο κατά την οποία μπορεί το ίδιο κλειδί να χρησιμοποιηθεί για την αποκρυπτογράφηση δεδομένων ή την επαλήθευση του MAC. Αυτό επιτρέπει στα δεδομένα που προστατεύονται από τον αποστολέα να υποβάλλονται σε επεξεργασία από τον παραλήπτη για μεγάλο χρονικό διάστημα μετά την εφαρμογή της προστασίας. Η (συνολική) κρυπτοπερίοδος ενός συμμετρικού κλειδιού είναι η χρονική περίοδος από την έναρξη της περιόδου χρήσης του από τον αποστολέα που εφαρμόζει την προστασία, έως τη λήξη της περιόδου χρήσης από τον παραλήπτη.

6.3.3 Κρυπτοπερίοδοι Ασύμμετρων Κλειδιών

Για ζεύγη κλειδιών ασύμμετρης κρυπτογραφίας, κάθε κλειδί του ζεύγους έχει τη δική του κρυπτοπερίοδο η οποία επηρεάζεται και από τη χρήση του κλειδιού. Πιο αναλυτικά:

- Για τα κλειδιά που χρησιμοποιούνται για τη δημιουργία και επαλήθευση ψηφιακών υπογραφών, η περίοδος χρήσης του ιδιωτικού κλειδιού (για τη δημιουργία υπογραφών) είναι συχνά μικρότερη από την περίοδο χρήσης του δημοσίου κλειδιού (για την επαλήθευση υπογραφών). Το ιδιωτικό κλειδί προορίζεται για χρήση για καθορισμένη χρονική περίοδο, μετά τη λήξη της οποίας ο κάτοχος του κλειδιού παύει να το κατέχει. Το δημόσιο κλειδί μπορεί και πρέπει να είναι διαθέσιμο τυπικά για πάντα, για την επαλήθευση των υπογραφών που δημιουργήθηκαν από το αντίστοιχο ιδιωτικό.

Όταν το ζεύγος κλειδιών χρησιμοποιείται για αυθεντικοποίηση οντότητας, π.χ. με την υπογραφή κάποιας πρόσκαιρης πρόκλησης, η κρυπτοπερίοδος του ιδιωτικού κλειδιού είναι ίδια με την κρυπτοπερίοδο του αντίστοιχου δημόσιου κλειδιού. Ως εκ τούτου, όταν το ιδιωτικό κλειδί δεν πρόκειται πλέον να χρησιμοποιείται για την υπογραφή προκλήσεων, το δημόσιο κλειδί δεν χρειάζεται πλέον.

- Για τα ασύμμετρα κλειδιά που χρησιμοποιούνται για τη μεταφορά κλειδιών συμμετρικής κρυπτογράφησης, το δημόσιο κλειδί χρησιμοποιείται για την κρυπτογράφηση των συμμετρικού κλειδιού και το αντίστοιχο ιδιωτικό για την αποκρυπτογράφηση των κρυπτογραφημένων δεδομένων/κλειδιών. Σε αυτή την περίπτωση, η περίοδος χρήσης του δημοσίου κλειδιού είναι συχνά μικρότερη από την περίοδο χρήσης του ιδιωτικού κλειδιού, καθώς παλαιότερες κρυπτογραφημένες πληροφορίες μπορεί να χρειάζεται να αποκρυπτογραφηθούν ακόμα και μετά την απόσυρση του δημοσίου κλειδιού.

6.4 Διαχείριση Συμμετρικών Κλειδιών

Η διαχείριση κλειδιών συμμετρικής κρυπτογραφίας έχει τυποποιηθεί στο διεθνές πρότυπο ISO/IEC 11770-2 [5] καθώς και σε τομεακά πρότυπα, όπως είναι το ISO 11568-2 [6] για τα χρηματοπιστωτικά ιδρύματα.

6.4.1 Δημιουργία και Διαμοιρασμό Συμμετρικών Κλειδιών

Τα συμμετρικά κλειδιά θα πρέπει:

1. να δημιουργούνται και στη συνέχεια να διαμοιράζονται χειροκίνητα, χρησιμοποιώντας έναν μηχανισμό μεταφοράς κλειδιού βασισμένο στη χρήση κρυπτογραφίας δημοσίου κλειδιού, ή με τη χρήση κάποιου κλειδιού περιτύλιξης (key wrapping), δηλαδή, συμμετρικού κλειδιού που χρησιμοποιείται για την κρυπτογράφηση άλλων κλειδιών, ή άλλου κλειδιού το οποίο διαμοιράστηκε στο παρελθόν, ή
2. να εδραιώνονται (βλέπε Ενότητα 6.5) χρησιμοποιώντας ένα πρωτόκολλο εδραίωσης ή συμφωνίας κλειδιών στο πλαίσιο του οποίου γίνεται η δημιουργία και ο διαμοιρασμός των κλειδιών, ή
3. να παράγονται από ένα κύριο κλειδί.

6.4.1.1 Δημιουργία Συμμετρικών Κλειδιών

Ένα συμμετρικό κλειδί μπορεί να δημιουργηθεί με χρήση μιας εγκεκριμένης γεννήτριας τυχαίων αριθμών [7], αλλά και από ένα κύριο κλειδί (master key) χρησιμοποιώντας μια μέθοδο παραγωγής κλειδιών (key derivation function) [8]. Συμμετρικά κλειδιά μπορούν επίσης να δημιουργηθούν χρησιμοποιώντας τεχνικές συμφωνίας κλειδιών [9, 10]. Σε αυτήν την περίπτωση, δεν απαιτείται ξεχωριστή διαδικασία διαμοιρασμού κλειδιού.

Κατά τη δημιουργία ενός συμμετρικού κλειδιού υπάρχει η δυνατότητα να χρησιμοποιηθούν διαδικασίες διαχωρισμού-γνώσης (split-knowledge) όπου το κλειδί που θα δημιουργηθεί, ενδεχομένως σε μια κρυπτογραφική μονάδα, θα χωριστεί σε ξεχωριστά μερίδια (shares). Δεν προκύπτει κάποια γνώση για το κλειδί

από κάποιο μερίδιο (π.χ. κάθε μερίδιο κλειδιού πρέπει να φαίνεται ότι δημιουργείται τυχαία). Εάν απαιτείται γνώση k μεριδίων για την κατασκευή ενός κλειδιού, τότε η γνώση οποιωνδήποτε $k - 1$ μεριδίων δεν θα παρέχει καμία πληροφορία για το κλειδί εκτός, ενδεχομένως, από το μήκος του. Αξίζει να σημειωθεί ότι η απλή συνένωση (π.χ. η δημιουργία ενός κλειδιού 128 bit με τη συνένωση δύο κλειδών 64 bit) δε θεωρείται κατάλληλη μέθοδος δημιουργίας κλειδιού. Τα συστήματα αυτά ονομάζονται συστήματα κοινής χρήσης κλειδιών και μελετώνται στην Ενότητα 6.6.

6.4.1.2 Διαμοιρασμός Κλειδιών

Τα κλειδιά που δημιουργούνται ως κλειδιά περιτύλιξης κλειδιών ή ως κύρια κλειδιά παραγωγής κλειδιού διαμοιράζονται χειροκίνητα (χειροκίνητο διαμοιρασμό κλειδιών) ή αυτοματοποιημένα με χρηση ενός πρωτοκόλλου μεταφοράς κλειδιών (αυτοματοποιημένο διαμοιρασμό κλειδιών).

Τα κλειδιά που χρησιμοποιούνται μόνο για την προστασία αποθηκευμένων πληροφοριών τυπικά δε διαμοιράζονται, εκτός αν πρόκειται για εφεδρικά κλειδιά ή για κλειδιά που πρέπει να δοθούν σε άλλες εξουσιοδοτημένες οντότητες που έχουν το δικαίωμα πρόσβασης στις αποθηκευμένες πληροφορίες που προστατεύονται με αυτά τα κλειδιά.

- Χειροκίνητος διαμοιρασμός κλειδιού:** Τα κλειδιά και τα μερίδια κλειδιών που διαμοιράζονται χειροκίνητα θα πρέπει να προστατεύονται σε όλη τη διαδικασία διαμοιρασμού. Τα συμμετρικά κλειδιά, τα ιδιωτικά κλειδιά και τα μερίδια κλειδιών θα πρέπει είτε να διανέμονται σε κρυπτογραφημένη μορφή, με ταυτόχρονη προστασία εμπιστευτικότητας και ακεραιότητας, είτε να διανέμονται με χρήση κατάλληλων διαδικασιών παροχής φυσικής ασφάλειας..

Η διαδικασία για τον χειροκίνητο διαμοιρασμό συμμετρικών κλειδιών, ιδιωτικών κλειδιών και μεριδίων κλειδιών θα πρέπει να διασφαλίζει ότι:

- Τα κλειδιά διαμοιράζονται από εξουσιοδοτημένη πηγή.
- Το κλειδιά προστατεύονται κατάλληλα καλύπτοντας τις απαιτήσεις που αναφέρθηκαν στην Ενότητα 6.2.
- Τα κλειδιά παραλαμβάνονται από τους εξουσιοδοτημένους παραλήπτες.
- Αυτοματοποιημένος διαμοιρασμός κλειδιού:** Μπορεί να χρησιμοποιηθεί για τον διαμοιρασμό συμμετρικών κλειδιών, ιδιωτικών κλειδιών και μεριδίων κλειδιών μέσω ενός ασφαλούς καναλιού επικοινωνίας. Αυτό απαιτεί το διαμοιρασμό/καθιέρωση ενός κλειδιού περιτύλιξης κλειδιού, δηλαδή ενός κλειδιού κρυπτογράφησης κλειδιού, ή ενός δημόσιου κλειδιού μεταφοράς.

6.5 Πρωτόκολλα Εδραίωσης Κλειδιών

Τα πρωτόκολλα εδραίωσης κλειδιών (key establishment protocols) έχουν σχεδιαστεί για τη δημιουργία συμμετρικών κλειδιών μεταξύ δύο ή περισσότερων πλευρών με ασφαλή τρόπο και με παράλληλη αυθεντικοποίηση (μονόδρομη ή αμφίδρομη) των δύο πλευρών που συμμετέχουν στην εκτέλεση του πρωτοκόλλου. Ο κύριος στόχος αυτών των πρωτοκόλλων είναι να διασφαλίσουν ότι τα μέρη που επικοινωνούν μπορούν να συμφωνήσουν σε ένα ή περισσότερα συμμετρικά κλειδιά χωρίς τον κίνδυνο υποκλοπής από τρίτους, μη εξουσιοδοτημένους χρήστες. Τα κλειδιά αυτά είναι συνήθως κλειδιά συνεδρίας και χρησιμοποιούνται για την προστασία των επικοινωνιών μεταξύ των πλευρών εξασφαλίζοντας έτσι ένα ασφαλές κανάλι επικοινωνίας.

Υπάρχουν δύο βασικές κατηγορίες πρωτόκολλων εδραίωσης κλειδιών. Αυτά που αφορούν στην συμφωνία κλειδιών (key agreement protocols) και αυτά που αφορούν στην μεταφορά κλειδιών (key transport protocols). Αυτά θα αναλυθούν στις ακόλουθες ενότητες παρέχοντας και σχετικά παραδείγματά τους. Πριν από αυτά ωστόσο, λίγα λόγια για τα κλειδιά συνεδρίας.

6.5.1 Κλειδιά Συνεδρίας

Σε μια επικοινωνία μεταξύ δύο οντοτήτων, πριν από τη φάση της ανταλλαγής μηνυμάτων, και εάν δεν έχουν διανεμηθεί ήδη κοινά κλειδιά, οι οντότητες που επικοινωνούν πρέπει να καθορίσουν τα κλειδιά που θα χρησιμοποιήσουν για την προστασία των ανταλλασσόμενων δεδομένων. Αυτά, τα λεγόμενα κλειδιά συνεδρίας ή συνόδου ή εφήμερα, αφορούν συνήθως κλειδιά για αλγορίθμους συμμετρικής κρυπτογραφίας και χρησιμοποιούνται για την προστασία δεδομένων στις επικοινωνίες (κρυπτογράφηση ή δημιουργία MAC). Πρόκειται για κλειδιά που έχουν μικρή διάρκεια ζωής, ίσως μερικά δευτερόλεπτα ή μια μέρα. Συνήθως χρησιμοποιούνται για την παροχή εμπιστευτικότητας για τη δεδομένη χρονική περίοδο. Η μη ξουσιοδοτημένη αποκάλυψη ενός κλειδιού συνεδρίας θα πρέπει να έχει ως αποτέλεσμα μόνο την παραβίαση του απορρήτου αυτής της περιόδου σύνδεσης και δεν θα πρέπει να επηρεάζει τη μακροπρόθεσμη ασφάλεια του συστήματος.

Η δημιουργία κλειδιών συνεδρίας υιοθετεί είτε ένα πρωτόκολλο μεταφοράς κλειδιών, όπου η μία εκ των δύο πλευρών δημιουργεί ένα κλειδί και το στέλνει στην άλλη πλευρά, είτε ένα πρωτόκολλο συμφωνίας, όπου και οι δύο πλευρές συμβάλλουν στη δημιουργία του κοινού κλειδιού και καμία από αυτές δεν έχει πλήρη έλεγχο της διαδικασίας δημιουργίας του κλειδιού. Τα κλειδιά συνεδρίας στη συνέχεια χρησιμοποιούνται, για παράδειγμα, από αλγορίθμους κρυπτογράφησης ή αυθεντικοποίησης δεδομένων.

Υπάρχουν διάφοροι λόγοι για τους οποίους οι δύο οντότητες θα πρέπει να χρησιμοποιούν κλειδιά ή κλειδιά συνεδρίας. Μεταξύ αυτών είναι ο περιορισμός τους αριθμού των μηνυμάτων που κρυπτογραφούνται με το ίδιο κλειδί και μπορεί να χρησιμοποιηθούν από τρίτους για κρυπτανάλυση. Επιπλέον, στην περίπτωση διαρροής ενός κλειδιού ο όγκος των δεδομένων που θα ποκρυπτογραφηθούν από μη ξουσιοδοτημένη οντότητα θα είναι περιορισμένος. Επίσης, τα κλειδιά δε χρειάζεται να αποθηκεύονται για μεγάλα χρονικά διαστήματα και να εκτίθενται σε πιθανή μη ξουσιοδοτημένη πρόσβαση.

6.5.2 Πρωτόκολλα Συμφωνίας Κλειδιού

Ένα πρωτόκολλο συμφωνίας κλειδιού (key agreement protocol) επιτρέπει σε δύο μέρη να συμφωνήσουν σε ένα ή περισσότερα κλειδιά που θα χρησιμοποιήσουν τυπικά για να προστατεύσουν τα μηνύματα που πρόκειται να ανταλλάξουν μεταξύ τους, δημιουργώντας έτσι ένα ασφαλές κανάλι επικοινωνίας. Οι απαιτήσεις ασφαλείας των πρωτοκόλλων συμφωνίας κλειδιού είναι πολύ υψηλές, οδηγώντας σε διάφορες ιδιότητες ασφαλείας που κάποια πρωτόκολλα που έχουν αναπτυχθεί, μπορεί να μη διαθέτουν [11]. Οι βασικές απαιτήσεις είναι οι εξής:

- Το κλειδί που δημιουργείται ως αποτέλεσμα της ορθής εκτέλεσης ενός πρωτοκόλλου συμφωνίας κλειδιού, θα πρέπει για κάποιον επιτιθέμενο να αποτελεί μια τυχαία τιμή, δηλαδή να μην αποκαλύπτει καμία πληροφορία που θα μπορούσε να βοηθήσει τον επιτιθέμενο να αναπαράξει το κλειδί.
- Κάθε συμμετέχουσα οντότητα είναι βέβαιη πως μόνο η άλλη πλευρά έχει πρόσβαση στο κλειδί που δημιουργήθηκε.

Άλλες ιδιότητες που απαιτούνται από τα σύγχρονα πρωτόκολλα συμφωνίας κλειδιών είναι οι εξής:

- Επιβεβαίωση κλειδιού (key confirmation): Κάθε μέρος είναι βέβαιο ότι το άλλο μέρος γνωρίζει το κοινό κλειδί. Μερικές φορές αυτή η απαίτηση μπορεί να εξαλειφθεί καθώς η μεταγενέστερη ορθή χρήση του κοινού κλειδιού παρέχει έμμεσα αυτήν την επιβεβαίωση. Αυτή η μεταγενέστερη διαδικασία ονομάζεται σιωπηρή επιβεβαίωση κλειδιού (implicit key confirmation).
- Προωθημένη μυστικότητα (forward secrecy): Η έκθεση ενός μακροπρόθεσμου συμμετρικού κλειδιού (long-term secret key) σε κάποια χρονική στιγμή στο μέλλον, δεν θέτει σε κίνδυνο την ασφάλεια κλειδιών που έχουν δημιουργηθεί από το μακροπρόθεσμο συμμετρικό κλειδί, και κατ'επέκταση των δεδομένων που έχουν προστατευτεί από αυτά στο παρελθόν.

Παραλλαγές στα πρωτόκολλα συμφωνίας κλειδιού περιλαμβάνουν τη συμφωνία κλειδιού ομάδας (group key agreement), η οποία επιτρέπει σε μια ομάδα χρηστών να συμφωνήσουν σε ένα κλειδί, καθώς και τη συμφωνία κλειδιού βασισμένη σε συνθηματικό (password-based key agreement), στην οποία δύο μέρη συμφωνούν σε ένα κλειδί μόνο εάν συμφωνούν επίσης σε κοινό συνθηματικό.

6.5.2.1 Πρωτόκολλο Diffie-Hellman

Το πρωτόκολλο Diffie-Hellman είναι ένα πρωτοκόλλο συμφωνίας κοινού κλειδιού [12, 13]. Υιοθετείται από έναν αριθμό ευρέως χρησιμοποιούμενων πρωτοκόλλων ασφαλούς μεταφοράς δεδομένων, όπως:

- Transport Layer Security – TLS (Ενότητα 10.2)
- Internet Protocol Security – IPsec (Ενότητα 10.3)
- Secure Shell – SSH (Ενότητα 10.4)

Από το πρωτόκολλο αυτό προέκυψε η ιδέα για την κρυπτογραφία δημοσίου κλειδιού. Απαιτεί από κάθε χρήστη να έχει ένα ζεύγος κλειδιών (ιδιωτικό-δημόσιο). Το δημόσιο κλειδί είναι πιστοποιημένο από έναν πάροχο υπηρεσιών εμπιστοσύνης και το ιδιωτικό κλειδί διατηρείται μυστικό από τον κάτοχο του ζεύγους κλειδιών. Η ανταλλαγή δημόσιων κλειδιών μεταξύ των δύο οντοτήτων, επιτρέπει στις δύο οντότητες κάνοντας χρήση των ιδιωτικών τους κλειδιών, να υπολογίσουν ένα κοινό μυστικό κλειδί. Εάν ο Μιχάλης, γενικότερα γνωστός στην κρυπτογραφία ως Mallory¹, παρακολουθήσει τις επικοινωνίες που πραγματοποιούνται μεταξύ της Αλίκης και του Μπάμπη, δεν θα είναι σε θέση να υπολογίσει το ίδιο κοινό κλειδί, καθώς αυτό απαιτεί τη γνώση τουλάχιστον ενός από τα δύο εμπλεκόμενα ιδιωτικά κλειδιά.

Για τις ανάγκες της επιτυχούς εκτέλεσης του πρωτοκόλλου, η Αλίκη και ο Μπάμπης θα πρέπει να συμφωνήσουν σε έναν μεγάλο πρώτο αριθμό p και έναν ακέραιο g τέτοιο ώστε, για οποιοδήποτε ακέραιο αριθμό n στο διάστημα $[1, p - 1]$, υπάρχει αριθμός k τέτοιος ώστε $g^k \equiv n \pmod{p}$.

Η Αλίκη επιλέγει έναν τυχαίο αριθμό $n < p$, και αντίστοιχα ο Μπάμπης έναν τυχαίο αριθμό $m < p$. Η Αλίκη υπολογίζει το δημόσιο κλειδί $g^n \pmod{p}$ και το στέλνει στον Μπάμπη. Ο Μπάμπης υπολογίζει και στέλνει το δικό του δημόσιο κλειδί $g^m \pmod{p}$ στην Αλίκη.

Η Αλίκη μπορεί τώρα να υπολογίσει το κοινό κλειδί:

$$s = (g^m)^n \pmod{p} = g^{mn} \pmod{p}$$

Κατά τον ίδιο τρόπο και ο Μπάμπης υπολογίζει το κοινό κλειδί:

$$s = (g^n)^m \pmod{p} = g^{mn} \pmod{p}$$

Έχοντας γνώση του κοινού μυστικού κλειδιού s , η Αλίκη μπορεί να το χρησιμοποιήσει ώστε να προστατέψει τα μηνύματα που στέλνει στον Μπάμπη. Ο Μπάμπης που επίσης γνωρίζει το κοινό κλειδί s , μπορεί να αποκρυπτογραφήσει κατάλληλα αυτά τα μηνύματα. Στο μεταξύ, κάποιος επιτιθέμενος που παρακολουθεί τις επικοινωνίες μεταξύ της Αλίκης και του Μπάμπη και υποκλέπτει τα $g^n \pmod{p}$ και $g^m \pmod{p}$, δε μπορεί να χρησιμοποιήσει αυτά τα μηνύματα ώστε να υπολογίσει κάποιο από τα $m, n, ή g^{mn} \pmod{p}$.

Το πρωτόκολλο Diffie-Hellman, σε αντίθεση με τη χρήση του RSA για μεταφορά εφήμερων κλειδιών², παρέχει προώθηση μυστικότητας, αλλά δεν παρέχει καμία μορφή ελέγχου ταυτότητας. Εξαιτίας αυτού, το πρωτόκολλο είναι ευάλωτο σε επιθέσεις man-in-the-middle οι οποίες μπορούν να εξελιχθούν ως εξής:

¹Είναι ο εισβολέας ή ο επιτιθέμενος που προσπαθεί να ανακτήσει ή να διαταράξει τις επικοινωνίες μεταξύ της Αλίκης και του Μπάμπη. Συνήθως αναπαρίσταται ως εχθρική οντότητα που επιδιώκει να αποκτήσει πρόσβαση σε πληροφορίες που δεν του ανήκουν.

²Ο RSA χρησιμοποιείται συχνά για την ασφαλή μεταφορά εφήμερων κλειδιών ή κλειδιών συνεδρίασης σε διάφορα πρωτόκολλα ασφαλείας, όπου ο αποστολέας κρυπτογραφεί το κλειδί συνεδρίασης με το δημόσιο κλειδί του παραλήπτη χρησιμοποιώντας τον αλγόριθμο RSA (βλέπε Ενότητα 6.5.3).

- Ο Μιχάλης, ως επιτιθέμενος, παρεμβάλλει στην επικοινωνίας μεταξύ της Αλίκης και του Μπάμπη.
- Όταν η Αλίκη στέλνει το δημόσιο κλειδί της $A = g^a \text{ mod } p$ στον Μπάμπη, ο Μιχάλης το αναχαιτίζει και στέλνει το δικό του δημόσιο κλειδί $M_A = g^{m_A} \text{ mod } p$ στον Μπάμπη αντί για το δημόσιο κλειδί της Αλίκης.
- Όταν ο Μπάμπης στέλνει το δημόσιο κλειδί του $B = g^b \text{ mod } p$ στην Αλίκη, ο Μιχάλης το αναχαιτίζει και στέλνει το δικό του δημόσιο κλειδί $M_B = g^{m_B} \text{ mod } p$ στην Αλίκη αντί για το δημόσιο κλειδί του Μπάμπη.
- Η Αλίκη λαμβάνει το M_B (νομίζοντας ότι είναι το δημόσιο κλειδί του Μπάμπη) και υπολογίζει το κοινό μυστικό:

$$S_A = (M_B)^a \text{ mod } p$$

- Ο Μπάμπης λαμβάνει το M_A (νομίζοντας ότι είναι το δημόσιο κλειδί της Αλίκης) και υπολογίζει το κοινό μυστικό:

$$S_B = (M_A)^b \text{ mod } p$$

- Ο Μιχάλης έχει τώρα δύο μυστικά κλειδιά:

- Το κοινό μυστικό με την Αλίκη: $S_{MA} = (A)^{m_B} \text{ mod } p$.
- Το κοινό μυστικό με τον Μπάμπη: $S_{MB} = (B)^{m_A} \text{ mod } p$.

Αυτά τα δύο κλειδιά του επιτρέπουν να αποκρυπτογραφεί τις επικοινωνίες που θα λάβουν χώρα στη συνέχεια μεταξύ της Αλίκης και του Μπάμπη, σπάζοντας έτσι το ασφαλές κανάλι επικοινωνίας που οι δύο πλευρές εσφαλμένα θεωρούν πως έχουν δημιουργήσει.

Για να αποφευχθεί αυτή η επίθεση και να επιτευχθεί αμοιβαία αυθεντικοποίηση μεταξύ της Αλίκης και του Μπάμπη, θα μπορούσαν τα δημόσια κλειδιά g^a mod p και g^b mod p της Αλίκης και του Μπάμπη αντίστοιχα, να πιστοποιηθούν κατάλληλα από έναν πάροχο υπηρεσών εμπιστοσύνης. Αυτό αποτρέπει την επίθεση man-in-the-middle. Τα ψηφιακά πιστοποιητικά για κλειδιά Diffie-Hellman παρέχουν σε τρίτα μέρη αδιαμφισβήτητες διαβεβαιώσεις για τον κάτοχο του ζεύγους κλειδιών. Ωστόσο έχουν το μειονέκτημα πως τα κλειδιά σε αυτή την περίπτωση είναι στατικά, καθώς ο κάτοχος δε δημιουργεί νέο ζεύγος για κάθε συνεδρία. Ως αποτέλεσμα, η χρήση πιστοποιημένων Diffie-Hellman κλειδιών μεταξύ συγκεκριμένων οντοτήτων οδηγεί στη δημιουργία πάντα του ίδιου κλειδιού συνεδρίας, εκτός εάν:

1. στη δημιουργία του κλειδιού συνεδρίας χρησιμοποιηθεί και κάποια μοναδική (nonce) τιμή, ή
2. πιστοποιημένο κλειδί χρησιμοποιείται μόνο από τη μια πλευρά, ενώ η άλλη δημιουργεί δυναμικά κάποιο ζεύγος κλειδιού – σε αυτή την περίπτωση ωστόσο η αυθεντικοποίηση που προσφέρει το πρωτόκολλο είναι μονόδρομη.

Η μονόδρομη πιστοποιημένη έκδοση της συμφωνίας κλειδιού Diffie-Hellman είναι η προτιμώμενη μέθοδος συμφωνίας κλειδιού στις σύγχρονες υλοποίησεις του πρωτοκόλλου TLS και είναι η μόνη μέθοδος συμφωνίας κλειδιού που υποστηρίζεται από το TLS 1.3.

6.5.2.2 Kerberos

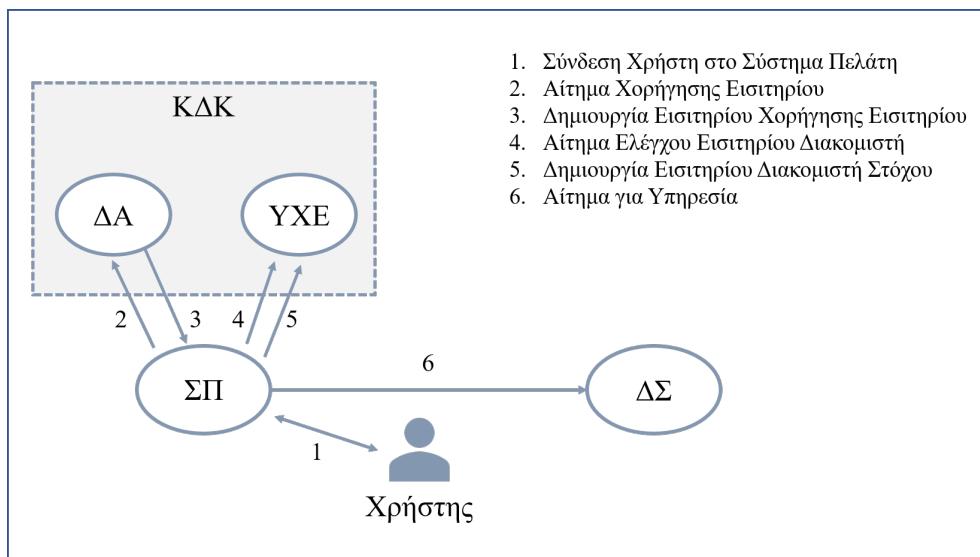
Το Kerberos είναι ένα σύστημα αυθεντικοποίησης που αναπτύχθηκε στο Ινστιτούτο Τεχνολογίας της Μασαχουσέτης (Massachusetts Institute of Technology – MIT) για να επιτρέψει την ασφαλή αυθεντικοποίηση των χρηστών σε Διακομιστές Στόχους – ΔΣ (Target Servers) μέσω ενός μη προστατευμένου καναλιού επικοινωνίας [14]. Βασίζεται σε συμμετρική κρυπτογραφία, στη χρήση διαμοιραζόμενων κλειδιών και ενός

διακομιστή αυθεντικοποίησης. Ο αρχικός σχεδιασμός και η υλοποίηση του Kerberos και των τριών πρώτων αναθεωρήσεών του (εκδόσεις 1 έως 4) ήταν κατά κύριο λόγο έργο των S.Miller, C.Neuman, J.Saltzer και J.Schiller. Μια τροποποιημένη έκδοση της αρχικής του Kerberos χρησιμοποιείται πλέον σε πολλές εκδόσεις του λειτουργικού συστήματος Windows και σε πολλά άλλα συστήματα.

Το Kerberos χρησιμοποιείται για τοπικές συνδέσεις, για απομακρυσμένο έλεγχο ταυτότητας (μέσω δικτύου), καθώς και για αιτήματα ενός Συστήματος Πελάτη (ΣΠ), ή απλά «πελάτη», προς έναν ΔΣ. Μπορεί επίσης να επεκταθεί ώστε να προβλέπει τη δημιουργία συμμετρικών κλειδιών μεταξύ ενός ΣΠ και ενός ΔΣ. Έχει σχεδιαστεί έτσι ώστε ένας χρήστης και ένας ΔΣ να βασίζονται σε μια αξιόπιστη τρίτη οντότητα για την αυθεντικοποίηση κάθε πλευράς.

Η έμπιστη τρίτη οντότητα είναι ένα Κέντρο Διαμοιρασμού Κλειδιών – ΚΔΚ (Key Distribution Center), το οποίο αποτελείται από έναν Διακομιστή Αυθεντικοποίησης – ΔΑ (Authentication Server) και μια Υπηρεσία Χορήγησης Εισιτηρίων – ΥΧΕ (Ticket Granting Service). Ο ΔΑ και η ΥΧΕ μπορεί να βρίσκονται στο ίδιο ή σε διαφορετικά συστήματα. Το ΚΔΚ διαθέτει μια βάση δεδομένων με συμμετρικά κλειδιά για τους χρήστες, τον ΔΣ και την ΥΧΕ. Όλα τα συμμετρικά κλειδιά του ΚΔΚ είναι προσβάσιμα από την ΥΧΕ.

Μια επισκόπηση της έκδοσης 5 του πρωτοκόλλου Kerberos φαίνεται στο Σχήμα 6.2. Στο σχήμα αυτό δεν απεικονίζονται λεπτομέρειες που αφορούν στη δημιουργία των κλειδιών συνόδου και στην προστασία των πληροφοριών που ανταλλάσσονται μεταξύ των εμπλεκομένων μερών. Για παράδειγμα, τα εισιτήρια και οι πληροφορίες αυθεντικοποίησης προστατεύονται με κώδικα αυθεντικοποίησης μηνύματος και κρυπτογράφηση όταν μεταδίδονται, κάτι που όμως δεν απεικονίζεται στο σχήμα.



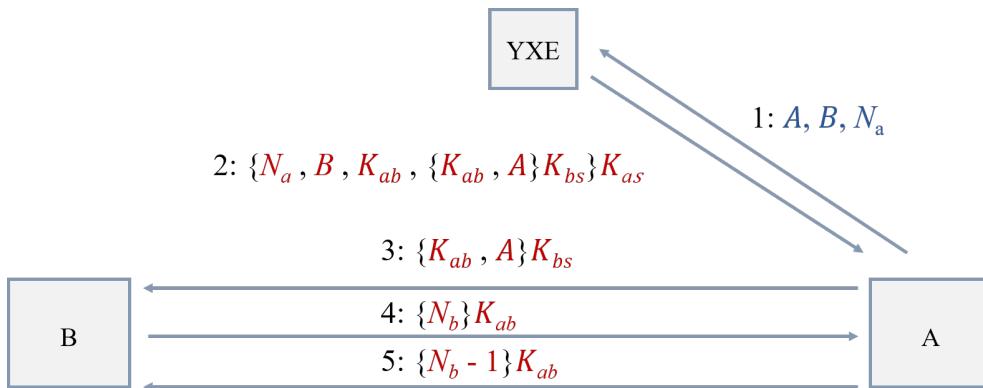
Το πρωτόκολλο αυθεντικοποίησης περιλαμβάνει τα ακόλουθα βήματα:

- Ο χρήστης συνδέεται σε ένα σύστημα-πελάτη εισάγοντας ένα συνθηματικό, από το οποίο δημιουργείται ένα συμμετρικό κλειδί χρήστη.
- Το σύστημα-πελάτης, ενεργώντας για λογαριασμό του χρήστη, ζητά Εισιτήριο Χορήγησης Εισιτηρίου – EXE (Ticket Granting Ticket) από τον ΔΑ. Ένα EXE είναι ένα διακριτικό αυθεντικοποίησης χρήστη (authentication token) που εκδίδεται από το ΚΔΚ και χρησιμοποιείται για να ζητήσει το σύστημα-πελάτης διακριτικά πρόσβασης από την ΥΧΕ για συγκεκριμένους πόρους/συστήματα που είναι συνδεδεμένα στον τομέα που ελέγχεται από το σύστημα Kerberos.
- Ο ΔΑ δημιουργεί ένα EXE, για μια καθορισμένη περίοδο ισχύος, και το στέλνει στο σύστημα-πελάτη.

4. Το σύστημα-πελάτη παρέχει το EXE στην YXE, μαζί με τις δικές του πληροφορίες αυθεντικοποίησης, οι οποίες περιλαμβάνουν το μοναδικό αναγνωριστικό του χρήστη και μια χρονοσήμανση.
5. Η YXE ελέγχει τις πληροφορίες ελέγχου ταυτότητας και την περίοδο ισχύος του EXE. Στη συνέχεια, η YXE δημιουργεί ένα Εισιτήριο Διακομιστή Στόχου – ΕΔΣ (Target Server Ticket) και το στέλνει στο σύστημα=πελάτη.
6. Το συστημα-πελάτη στέλνει πληροφορίες αυθεντικοποίησης και το ΕΔΣ στον ΔΣ.
7. Ο ΔΣ ελέγχει τις πληροφορίες αυθεντικοποίησης και την περίοδο ισχύος του Εισιτηρίου Διακομιστή Στόχου. Εάν οι πληροφορίες είναι έγκυρες, ο χρήστης αυθεντικοποιείται από τον ΔΣ.

Κοιτώντας λίγο πιο αναλυτικά το πρωτόκολλο Kerberos, και πιο συγκεκριμένα, τη διαδικασία χορήγησης εισιτήριων από την YXE για πρόσβαση σε κάποιον ΔΣ, ας υποθέσουμε ότι ο A επιθυμεί να αποκτήσει πρόσβαση σε έναν πόρο B σε κάποιον ΔΣ. Πρώτα ο A αυθεντικοποιείται από τον ΔΑ χρησιμοποιώντας τα κατάλληλα διαπιστευτήρια. Ο ΔΑ δίνει στον χρήστη A ένα EXE κρυπτογραφημένο με το συνθηματικό του. Αυτό το εισιτήριο περιέχει ένα κλειδί συνόδου K_{as} . Ο A τώρα χρησιμοποιεί το K_{as} για να αποκτήσει ένα εισιτήριο από την YXE με το οποίο θα αποκτήσει πρόσβαση στον πόρο B. Η απάντηση της YXE είναι ένα κλειδί K_{ab} , μια χρονοσήμανση TS και η διάρκεια ζωής του εισιτηρίου L . Η απάντηση της YXE χρησιμοποιείται για την αυθεντικοποίηση του A σε επακόλουθη επικοινωνία με τον B.

Μια απλουστευμένη ροή των μηνυμάτων κατά την εκτέλεση του Kerberos απεικονίζεται στο Σχήμα 6.3.



Σχήμα 6.3: Ανταλλαγή μηνυμάτων στο πρωτόκολλο Kerberos.

Η εκτέλεση των πρωτοκόλλου περιλαμβάνει την ανταλλαγή των ακόλουθων μηνυμάτων:

- Ο A ενημερώνει την YXE πως θέλει να αποκτήσει πρόσβαση στο B.
- Εάν η YXE επιτρέπει αυτήν την πρόσβαση, δημιουργείται έναν κλειδί συνόδου K_{ab} και ένα εισιτήριο $\{T_s, L, K_{ab}\}$, όπου L η διάρκεια ισχύος του εισιτηρίου (αποτελούμενη από ώρα έναρξης και λήξης). Αυτό κρυπτογραφείται με το K_{bs} και αποστέλλεται στον A για προώθηση στο B. Ο χρήστης A λαμβάνει επίσης ένα αντίγραφο του κλειδιού σε μια μορφή που μπορεί να διαβαστεί μόνο από αυτόν.
- Ο A επαληθεύει ότι το εισιτήριο είναι έγκυρο και ότι ο πόρος B είναι σε λειτουργία. Ως εκ τούτου, στέλνει ένα κρυπτογραφημένο nonce/χρονοσφραγίδα T_A στο B.
- Ο πόρος B στέλνει πίσω στον A κρυπτογραφημένο το $\{T_A + 1\}$, αφού ελέγχει ότι η χρονοσήμανση T_A είναι πρόσφατη, αποδεικνύοντας ότι γνωρίζει το κλειδί και ότι είναι σε λειτουργία.

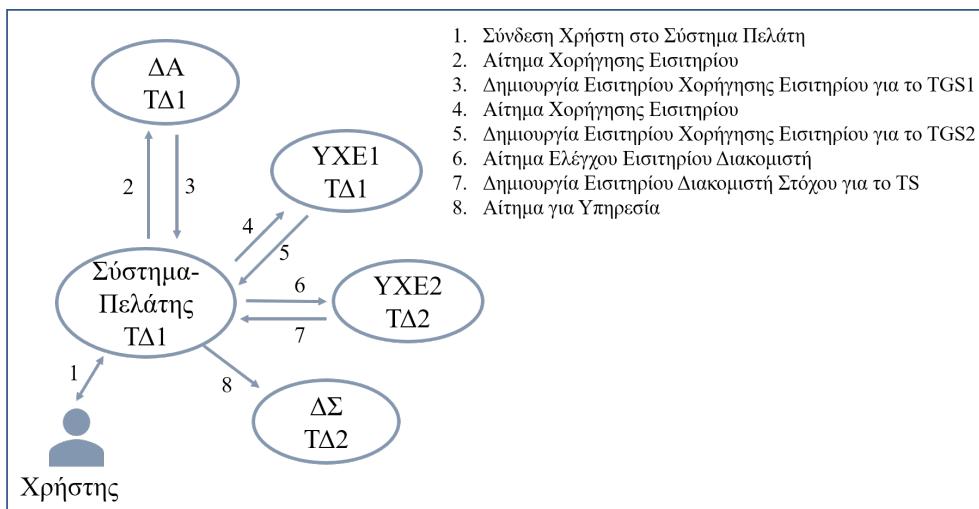
Σε μια παραλλαγή του πρωτοκόλλου μεταξύ του συστήματος-πελάτη και του ΔΑ, τα κλειδιά συνόδου δε δημιουργούνται από συνθηματικά, αλλά με τη χρήση δημοσίων κλειδιών όπου, είτε ο χρήστης και ο ΔΑ έχουν ζεύγη δημόσιων κλειδιών που χρησιμοποιούνται για εδραίωση συμμετρικών κλειδιών, είτε ο χρήστης έχει ένα ζεύγος κλειδιών εδραίωσης κλειδιών και ο ΔΑ διαθέτει ζεύγος κλειδιών ψηφιακής υπογραφής [15]. Σε αυτή την περίπτωση, το συμμετρικό κλειδί χρήστη μπορεί να δημιουργηθεί μεταξύ του συστήματος-πελάτη και του ΚΔΚ με έναν από τους ακόλουθους δύο τρόπους:

- Με τη χρήση συμφωνίας κλειδιού (π.χ. Diffie-Hellman) μεταξύ του ΔΑ και του πελάτη, ή
- Με τη μεταφορά κλειδιού (π.χ. RSA), όπου ο ΔΑ δημιουργεί το συμμετρικό κλειδί χρήστη και στέλνει το κλειδί στον πελάτη, κρυπτογραφημένο με το δημόσιο κλειδί του.

Για την προστασία όλης της επικοινωνίας μεταξύ του συστήματος-πελάτη και του ΚΔΚ, μπορεί να χρησιμοποιηθεί και το TLS πρωτόκολλο, όπως περιγράφεται στο RFC 6251 [16].

Επιπλέον, το πρωτόκολλο μπορεί να επεκταθεί για την αυθεντικοποίηση του ΔΣ από τον χρήστη. Επιπλέον, ένα εισιτήριο μπορεί να επαναχρησιμοποιηθεί εντός της περιόδου ισχύος του.

Τομείς Διαχείρισης: Κάθε YXE έχει το δικό της Τομέα Διαχείρισης – TΔ, γνωστό και ως βασίλειο (realm) συστημάτων-πελατών και διακομιστών-στόχων. Ένα βασίλειο είναι ένα λογικό δίκτυο που περιλαμβάνει μια ομάδα συστημάτων κάτω από το ίδιο κύριο ΚΔΚ. Ωστόσο, διαφορετικοί TΔ μπορεί να συνδέονται με την κοινή χρήση δια-τομεακών κλειδιών μεταξύ YXE, όπως απεικονίζεται στο Σχήμα 6.4. Ένα σύστημα-πελάτης στον TΔ1 που επιθυμεί πρόσβαση σε μια υπηρεσία ενός ΔΣ στον TΔ2 μπορεί να λάβει ένα εισιτήριο από το EXE1 που εισάγει το σύστημα-πελάτη στο EXE2. Αυτό το εισιτήριο είναι κρυπτογραφημένο με το διατομεακό κλειδί που μοιράζονται οι EXE1 και EXE2. Στη συνέχεια, το σύστημα-πελάτης μπορεί να ζητήσει ένα εισιτήριο από τον EXE2 για την επιθυμητή υπηρεσία στο ΔΣ στον TΔ2. Έτσι, οι TΔ μπορεί να είναι δικτυωμένοι για να παρέχουν στους πελάτες δια-τομεακές υπηρεσίες.



Σχήμα 6.4: Επισκόπηση του πρωτοκόλλου Kerberos με σύνδεση διαφορετικών τομέων διαχείρισης.

6.5.3 Πρωτόκολλα Μεταφοράς Κλειδιών

Σε ένα πρωτόκολλο μεταφοράς κλειδιού το ένα μέρος δημιουργεί ή αποκτά με κάποιον τρόπο μια μυστική τιμή και τη μεταφέρει με ασφάλεια στο άλλο. Τα πρωτόκολλα μεταφοράς κλειδιού συνήθως κάνουν χρήση κρυπτογραφίας δημόσιου κλειδιού. Για παράδειγμα, στην μεταφορά κλειδιού με RSA, ο αποστολέας κρυπτογραφεί

ένα κλειδί συνεδρίας με το δημόσιο κλειδί του παραλήπτη και το στέλνει στην άλλη πλευρά, όπου αποκρυπτογραφείται χρησιμοποιώντας το ιδιωτικό κλειδί του παραλήπτη.

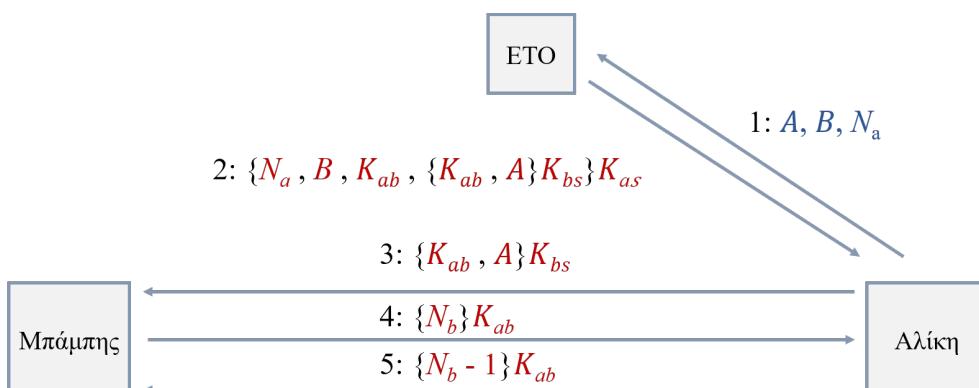
Η μεταφορά κλειδιού μέσω κρυπτογράφησης δημόσιου κλειδιού δεν εξασφαλίζει προώθηση μυστικότητας (forward secrecy). Για να γίνει καλύτερα αντιληπτό γιατί αυτό είναι σημαντικό, ας υποθέσουμε ότι μια πλευρά κρυπτογραφεί μαζικά μια ροή βίντεο και, στη συνέχεια, κρυπτογραφεί το κλειδί συνεδρίας με τη χρήση του δημόσιου κλειδιού RSA του παραλήπτη. Ας υποθέσουμε τώρα ότι κάποια στιγμή στο μέλλον, το ιδιωτικό κλειδί RSA του παραλήπτη παραβιάζεται. Σε μια τέτοια περίπτωση, όλη η ροή του βίντεο μπορεί επίσης να αποκρυπτογραφηθεί από την οντότητα που αποκτά πρόσβαση στο ιδιωτικό κλειδί, καθώς αυτό θα δώσει πρόσβαση και στα κλειδιά συνεδρίας που έχουν κρυπτογραφηθεί με το αντίστοιχο δημόσιο κλειδί. Όλο αυτό βέβαια με την προϋπόθεση ότι ο επιτιθέμενος κατέγραψε ολόκληρη τη σύνοδο.

Επιπλέον, η χρήση της μεταφοράς κλειδιού σημαίνει ότι ο παραλήπτης εμπιστεύεται τον αποστολέα ότι μπορεί να δημιουργήσει ένα ισχυρό κλειδί συνεδρίας, το οποίο μπορεί να καλύψει επαρκώς τις ανάγκες προστασίας της επικοινωνίας. Για να είναι απόλυτα σίγουρος ο παραλήπτης μπορεί να θέλει να συνεισφέρει κάποια δική του τυχαιότητα στο κλειδί συνεδρίας (πρωτόκολλο συμφωνίας κλειδιού). Αυτό μπορεί να γίνει μόνο εάν και τα δύο μέρη είναι ενεργά και συνδεδεμένα στο διαδίκτυο τη χρονική στιγμή της εκτέλεσης του πρωτοκόλλου εδραίωσης κλειδιού. Η μεταφορά κλειδιού είναι πιο κατάλληλη για την περίπτωση που μόνο ο αποστολέας είναι online, όπως για παράδειγμα σε εφαρμογές όπως το email.

6.5.3.1 Πρωτόκολλο Needham-Schroeder

Το πρωτόκολλο Needham–Schroeder [17] χρησιμοποιεί συμμετρική κρυπτογραφία. Είναι ένα από τα δύο βασικά πρωτόκολλα μεταφοράς κλειδιών που προτάθηκαν από τους Roger Needham και Michael Schroeder, με το άλλο να βασίζεται στη χρήση κρυπτογραφίας δημόσιου κλειδιού. Αναπτύχθηκε το 1978 και είναι ένα από τα πιο μελετημένα πρωτόκολλα. Η φήμη του οφείλεται στο γεγονός ότι ακόμη και ένα απλό πρωτόκολλο μπορεί να κρύψει ελαττώματα ασφαλείας για μεγάλο χρονικό διάστημα. Αποτελεί τη βάση του πρωτοκόλλου Kerberos (βλέπε Ενότητα 6.5.2.2).

Στο Σχήμα 6.5 απεικονίζονται τα μηνύματα που ανταλλάσσονται κατά την εκτέλεση του πρωτοκόλλου. Ας υποθέσουμε ότι τα δύο μέρη που επιθυμούν να συμφωνήσουν ένα μυστικό είναι η Αλίκη και ο Μπάμπης οι οποίες στο σύστημα είναι γνωστές ως A και B. Υποθέτουμε ότι και οι δύο χρησιμοποιούν τις υπηρεσίες μιας Έμπιστης Τρίτης Οντότητας – ETO (Trusted Third Party).



Σχήμα 6.5: Πρωτόκολλο Needham-Schroeder.

Συγκεκριμένα:

- Η Αλίκη δημιουργεί μια τυχαία τιμή nonce N_a και ενημερώνει με το πρώτο μήνυμα την ETO πως θέλει ένα κλειδί για να επικοινωνήσει με τον Μπάμπη.
- Στο δεύτερο μήνυμα, η ETO δημιουργεί το κλειδί συνόδου K_{ab} το οποίο θα μοιραστούν η Αλίκη με τον

Μπάμπη και το στέλνει πίσω στην Αλίκη κρυπτογραφημένο με το κλειδί K_{as} το οποίο μοιράζονται η Αλίκη με την ΕΤΟ. Η μοναδική τυχαία τιμή (nonce) N_a περιλαμβάνεται στο μήνυμα έτσι ώστε η Αλίκη να γνωρίζει ότι αυτό στάλθηκε ως απάντηση στο πρώτο μήνυμά της. Επιπλέον, το κλειδί συνόδου K_{ab} κρυπτογραφείται με το κλειδί K_{bs} το οποίο μοιράζεται η ΕΤΟ με τον Μπάμπη για αποστολή του K_{ab} στον Μπάμπη.

- Με το τρίτο μήνυμα, η Αλίκη προωθεί το κλειδί συνόδου K_{ab} στον Μπάμπη.
- Ο Μπάμπης πρέπει να ελέγξει ότι το τρίτο μήνυμα δεν ήταν επανάληψη από προηγούμενη εκτέλεση του πρωτοκόλλου. Πρέπει λοιπόν να επιβεβαιώσει ότι η Αλίκη συμμετέχει σε αυτή τη συνομιλία, και έτσι, στο τέταρτο μήνυμα κρυπτογραφεί με το κλειδί K_{ab} ένα nonce N_b και το στέλνει στην Αλίκη.
- Στο τελευταίο μήνυμα, για να αποδείξει η Αλίκη στον Μπάμπη πως είναι ενεργή, κρυπτογραφεί μια απλή συνάρτηση του nonce του Μπάμπη (π.χ. $N_a - 1$ και στέλνει το αποτέλεσμα πίσω στον Μπάμπη.

Το κύριο πρόβλημα με το πρωτόκολλο Needham–Schroeder είναι ότι ο Μπάμπης δε γνωρίζει ότι το κλειδί που μοιράζεται με την Αλίκη είναι νέο κλειδί, και όχι επαναχρησιμοποίηση παλιού κλειδιού, αδυναμία που εντοπίστηκε λίγο καιρό μετά τη δημοσίευση του αρχικού πρωτοκόλλου. Ένας επιτιθέμενος που θα βρει τα μηνύματα μιας παλιάς συνόδου (συνήθως ανταλλάσσονται μέσω δημοσίου δικτύου) και αν υποθέσουμε ότι θα μπορέσει με κάποιον τρόπο να βρει το κλειδί συνόδου που χρησιμοποιήθηκε, να χρησιμοποιήσει τα τρία τελευταία μηνύματα της προηγούμενης συνόδου που αφορούν στον Μπάμπη σε μια επίθεση επανάληψης. Επομένως, ο επιτιθέμενος μπορεί να πείσει τον Μπάμπη να συμφωνήσει με ένα κλειδί με τον επιτιθέμενο, που ο Μπάμπης πιστεύει ότι μοιράζεται με την Αλίκη.

Σημειώστε ότι η Αλίκη και ο Μπάμπης χρησιμοποιούν το κλειδί συνόδου K_{ab} που δημιουργείται από την ΕΤΟ και έτσι κανένα από τα δύο μέρη δεν χρειάζεται να εμπιστεύεται το άλλο για την παραγωγή ισχυρών κλειδιών.

6.6 Σχήματα Κοινής Χρήσης Μυστικών

Ας υποθέσουμε ότι έχετε ένα μυστικό s το οποίο θέλετε να μοιραστείτε μεταξύ n μερών από ένα σύνολο μερών P και ότι θα θέλατε συγκεκριμένα υποσύνολα των n μερών να μπορούν να ανακτήσουν το κοινό μυστικό άλλα όχι άλλα. Το κλασικό σενάριο μπορεί να είναι ότι το s είναι κώδικας εκτόξευσης μιας πυρηνικής κεφαλής και ότι υπάρχουν τέσσερα άτομα, ο πρόεδρος, ο αντιπρόεδρος, ο υπουργός Άμυνας και ένας στρατηγός σε ένα σιλό πυραύλων. Δεν θέλετε ο στρατηγός να μπορεί να εκτοξεύσει τον πύραυλο χωρίς να συμφωνήσει ο πρόεδρος, αλλά για την περίπτωση που ο πρόεδρος δεν είναι διαθέσιμος, ο αντιπρόεδρος, ο υπουργός Εξωτερικών και ο στρατηγός θα μπορούν να συμφωνήσουν να εκτοξεύσουν τον πύραυλο. Εάν χαρακτηρίσουμε τα τέσσερα μέρη ως Π, A, Γ και Σ , για Πρόεδρο, Αντιπρόεδρο, Υπουργό Άμυνας και Στρατηγό, τότε θα θέλαμε τα ακόλουθα σύνολα ανθρώπων να μπορούν να εκτοξεύσουν τον πύραυλο:

$$\{\Pi, \Sigma\} \text{ και } \{A, \Gamma, \Sigma\}$$

αλλά όχι μικρότερα σύνολα. Τα Σχήματα Κοινής Χρήσης Μυστικών (Secret sharing schemes), γνωστά και ως Σχήματα Διάσπασης Κλειδιού (key splitting schemes), έρχονται να δώσουν λύση σε τέτοιους είδους προβλήματα.

Σε κάθε οντότητα που συμμετέχει στο σχήμα, διαμοιράζονται κάποιες πληροφορίες που ονομάζονται μερίδια (shares). Για την οντότητα A θα θεωρήσουμε πως το s_A υποδηλώνει το δικό της μερίδιο. Στο παραπάνω παράδειγμα υπάρχουν τέσσερα τέτοια μερίδια s_Π, s_A, s_Γ και s_Σ . Στη συνέχεια, εάν τα απαιτούμενα μέρη συνεργαστούν, θα θέλαμε έναν αλγόριθμο που συνδυάζει τα σχετικά μερίδια τους στο μυστικό s .

Πριν την περιγραφή των σχημάτων κοινής χρήσης μυστικού, πρέπει να εισαγάγουμε την έννοια της δομής πρόσβασης. Οποιοδήποτε υποσύνολο των μερών (οντοτήτων που συμμετέχουν στο σχήμα) που μπορούν να

ανακτήσουν το μυστικό ονομάζεται κατάλληλο σύνολο (qualifying set), ενώ το σύνολο όλων των κατάλληλων συνόλων ονομάζεται δομή πρόσβασης. Έτσι στο παραπάνω παράδειγμα έχουμε ότι τα ακόλουθα δύο σύνολα είναι κατάλληλα σύνολα:

$$\{\Pi, \Sigma\} \text{ και } \{\Lambda, \Upsilon, \Sigma\}$$

Είναι σαφές ότι κάθε σύνολο που περιέχει ένα τέτοιο κατάλληλο σύνολο είναι επίσης κατάλληλο σύνολο. Έτσι, τα:

$$\{\Pi, \Sigma, \Lambda\}, \{\Pi, \Sigma, \Upsilon\} \text{ και } \{\Pi, \Lambda, \Upsilon, \Sigma\}$$

είναι επίσης κατάλληλα σύνολα. Ως εκ τούτου, υπάρχουν πέντε σύνολα στη δομή πρόσβασης. Για οποιοδήποτε κατάλληλο σύνολο στη δομή πρόσβασης, εάν έχουμε όλα τα μερίδια του συνόλου, θα πρέπει να μπορούμε να ανακατασκευάσουμε το μυστικό.

Ορισμός 6.1 (Μονότονη Δομή). Θεωρήστε ένα σύνολο P . Μια μονότονη δομή στο P είναι μια συλλογή Γ υποσυνόλων του P , έτσι ώστε:

- $P \in \Gamma$
- Εάν $A \in \Gamma$ και B είναι ένα σύνολο τέτοιο ώστε $A \subset B \subset P$, τότε $B \in \Gamma$

Έτσι στο παραπάνω παράδειγμα η δομή πρόσβασης είναι μονότονη. Αυτή είναι μια ιδιότητα που θα ισχύει για όλες τις δομές πρόσβασης όλων των σχημάτων κοινής χρήσης μυστικών. Για μια μονότονη δομή σημειώνουμε ότι τα υποσύνολα στο Γ σχηματίζουν αλυσίδες, $A \subset B \subset C \subset P$. Θα ονομάσουμε τα σύνολα στην αρχή μιας αλυσίδας, ελάχιστα κατάλληλα σύνολα. Το σύνολο όλων των ελάχιστων συνόλων για μια δομή πρόσβασης Γ θα το συμβολίσουμε με $m(\Gamma)$.

Για παράδειγμα, ας θεωρήσουμε το σύνολο $P = A, B, C, D$. Ας πούμε ότι η δομή πρόσβασης Γ περιέχει τα υποσύνολα $A, B, A, C, B, C, A, B, C$, και P . Για να βρούμε τα ελάχιστα κατάλληλα σύνολα, ψάχνουμε τα υποσύνολα της δομής πρόσβασης που δεν περιέχουν κανένα άλλο υποσύνολο της δομής πρόσβασης. Στην περίπτωσή μας, τα ελάχιστα κατάλληλα σύνολα είναι A, B, A, C και B, C . Αυτά είναι τα σύνολα που δεν περιέχουν κανένα άλλο σύνολο της Γ , αλλά κάθε σύνολο που περιέχει αυτά τα σύνολα ανήκει στη Γ . Το σύνολο όλων αυτών των ελάχιστων συνόλων για τη δομή πρόσβασης Γ , δηλ. τα A, B, A, C, B, C είναι το $m(\Gamma)$.

Μπορούμε τώρα να δώσουμε έναν ορισμό του τι εννοούμε ως σχήμα κοινής χρήσης μυστικού:

Ορισμός 6.2 (Σχήμα Κοινής Χρήσης Μυστικού). Ένα σχήμα κοινής χρήσης μυστικών για μια μονότονη δομή πρόσβασης Γ σε ένα σύνολο οντοτήτων P σε σχέση με ένα χώρο μυστικών S , είναι ένα ζεύγος αλγορίθμων που ονομάζεται *Share* και *Recombine* με τις ακόλουθες ιδιότητες:

- Ο αλγόριθμος $Share(s, \Gamma)$ παίρνει ένα μυστικό s και μια μονότονη δομή πρόσβασης Γ και καθορίζει μια τιμή s_A για κάθε κάθε οντότητα A που είναι μέλος του P ($A \in P$). Η τιμή s_A ονομάζεται μερίδιο του μυστικού της οντότητας A .
- Ο αλγόριθμος $Recombine(H)$ παίρνει ένα σύνολο H μεριδίων για κάποιο υποσύνολο οντοτήτων \mathcal{O} του P , δηλαδή:

$$H = \{s_O : O \in \mathcal{O}\}$$

Εάν $\mathcal{O} \in \Gamma$ τότε το $Recombine(H)$ θα πρέπει να επιστρέψει το μυστικό s , διαφορετικά δεν θα πρέπει να επιστρέψει τίποτα.

Ένα σύστημα κοινής χρήσης μυστικών θεωρείται ασφαλές εάν κανένας επιτιθέμενος δεν μπορεί να μάθει κάτι για το κοινό μυστικό χωρίς να έχει πρόσβαση στα μερίδια ενός κατάλληλου συνόλου.

Για τις ανάγκες της παρουσίασης των σχημάτων κοινής χρήσης μυστικών θα εξετάσουμε δύο παραδείγματα δομών μονότονης πρόσβασης, έτσι ώστε να επεξηγηθούν κατάλληλα τα σχήματα κοινής χρήσης μυστικών. Και τα δύο παραδείγματα είναι σε σύνολα τεσσάρων στοιχείων: Το πρώτο είναι από το παραπάνω παράδειγμα όπου έχουμε $P = \{\Pi, A, \Upsilon, \Sigma\}$ και:

$$\Gamma = \{\Pi, \Sigma\}, \{A, \Upsilon, \Sigma\}, \{\Pi, \Sigma, A\}, \{\Pi, \Sigma, \Upsilon\}, \{\Pi, A, \Upsilon, \Sigma\}$$

Το σύνολο των ελάχιστων κατάλληλων συνόλων δίνεται από το:

$$\{\Pi, \Sigma\}, \{A, \Upsilon, \Sigma\}$$

Για το δεύτερο παράδειγμα, θα ορίσουμε πάνω από το σύνολο των οντοτήτων $P = \{A, B, \Gamma, \Delta\}$, με δομή πρόσβασης:

$$\Gamma = \{A, B\}, \{A, \Gamma\}, \{A, \Delta\}, \{B, \Gamma\}, \{B, \Delta\}, \{\Gamma, \Delta\}, \{A, B, \Gamma\}, \{A, B, \Delta\}, \{B, \Gamma, \Delta\}, \{A, B, \Gamma, \Delta\}$$

Το σύνολο των ελάχιστων κατάλληλων συνόλων δίνεται από το:

$$\{A, B\}, \{A, \Gamma\}, \{A, \Delta\}, \{B, \Gamma\}, \{B, \Delta\}, \{\Gamma, \Delta\}$$

Αυτή η τελευταία δομή πρόσβασης είναι ενδιαφέρονσα καθώς απαιτούμε κάθε δύο από τα τέσσερα μέρη να μπορούν να ανακτήσουν το κοινό μυστικό. Ένα τέτοιο σχήμα ονομάζεται δομή πρόσβασης κατωφλίου 2-από-4.

Ένας τρόπος εξέτασης τέτοιων δομών πρόσβασης είναι μέσω ενός τύπου boolean. Στο πρώτο παράδειγμα που δόθηκε παραπάνω οι τύποι για τα ελάχιστα κατάλληλα σύνολα γίνονται:

$$\{\Pi \wedge \Sigma\} \vee \{A \wedge \Upsilon \wedge \Sigma\}$$

Διαβάζοντας αυτούς τους τύπους, με το \wedge να είναι το λογικό “AND” και το \vee να είναι το “OR”, βλέπουμε ότι μπορεί κανείς να ανασκευάσει το μυστικό εάν έχει πρόσβαση στα μυστικά μερίδια:

$$\{\Pi \text{ AND } \Sigma\} \text{ OR } \{A \text{ AND } \Upsilon \text{ AND } \Sigma\}$$

6.6.1 Σχήμα Κοινής Χρήσης Μυστικών Ito-Nishizeki-Saito

Το σχήμα κοινής χρήσης μυστικών Ito-Nishizeki-Saito χρησιμοποιεί τους boolean τύπους που παρουσιάστηκαν παραπάνω. Κάθε “OR” μετατρέπεται σε λειτουργία συνένωσης και κάθε “AND” μετατρέπεται σε λειτουργία XOR. Σημειώστε ότι αυτό μπορεί εκ πρώτης όψεως να φαίνεται ελαφρώς μη λογικό.

Ο αλγόριθμος αυτός λειτουργεί ως εξής. Για κάθε ελάχιστο κατάλληλο σύνολο $\mathcal{O} \in m(\Gamma)$, δημιουργούμε τυχαία μερίδια $s_i \in S$, για $1 \leq i \leq l$, τυχαία, όπου $l = |\mathcal{O}|$ τέτοια ώστε $s_1 \oplus \dots \oplus s_l = s$. Τότε σε μια οντότητα A δίνεται ένα μερίδιο s_i αν εμφανίζεται ως θέση i στο σύνολο \mathcal{O} .

Παράδειγμα 1: Σε αυτό το παράδειγμα χρησιμοποιούμε τους τύπους που ορίσαμε προηγουμένως:

$$(\Pi \text{ AND } \Sigma) \text{ OR } (A \text{ AND } \Upsilon \text{ AND } \Sigma)$$

Δημιουργούμε πέντε μερίδια s_i από το S έτσι ώστε:

$$\begin{aligned} s &= s_1 \oplus s_2, \\ &= s_3 \oplus s_4 \oplus s_5 \end{aligned}$$

Στη συνέχεια, τα τέσσερα μερίδια ορίζονται ως:

$$\begin{aligned}s_{\Pi} &= s_1 \\s_A &= s_3 \\s_{\Gamma} &= s_4 \\s_{\Sigma} &= s_2 \parallel s_5\end{aligned}$$

Δεδομένης αυτής της κοινής χρήσης, ελέγξτε πως οποιοδήποτε κατάλληλο σύνολο μπορεί να ανακτήσει το μυστικό και μόνο τα κατάλληλα σύνολα μπορούν να ανακτήσουν το μυστικό. Σημειώστε ότι η οντότητα Σ χρειάζεται να διατηρεί δύο φορές περισσότερα δεδομένα από το μέγεθος του μυστικού (s_2 και s_5). Επομένως, αυτό το σύστημα σε αυτή την περίπτωση δεν είναι ιδιαίτερα αποτελεσματικό. Στην ιδανική περίπτωση, θα θέλαμε οι συμμετέχουσες οντότητες να διατηρούν μόνο το ισοδύναμο n-bit πληροφοριών η καθεμία, για τις ανάγκες της ανάκτησης ενός μυστικού n-bit.

Παράδειγμα 2: Σε αυτό το παράδειγμα ο τύπος δίνεται από το:

$$(A \text{ AND } B) \text{ OR } (A \text{ AND } \Gamma) \text{ OR } (A \text{ AND } \Delta) \text{ OR } (B \text{ AND } \Gamma) \text{ OR } (B \text{ AND } \Delta) \text{ OR } (\Gamma \text{ AND } \Delta)$$

Τώρα δημιουργούμε μερίδια για δώδεκα στοιχεία s_i από το S , έτσι ώστε

$$\begin{aligned}s &= s_1 \oplus s_2, \\&= s_3 \oplus s_4, \\&= s_5 \oplus s_6, \\&= s_7 \oplus s_8, \\&= s_9 \oplus s_{10}, \\&= s_{11} \oplus s_{12}\end{aligned}$$

Τα τέσσερα μερίδια ορίζονται ως:

$$\begin{aligned}s_A &= s_1 \parallel s_3 \parallel s_5 \\s_B &= s_2 \parallel s_7 \parallel s_9 \\s_{\Gamma} &= s_4 \parallel s_8 \parallel s_{11} \\s_{\Delta} &= s_6 \parallel s_{10} \parallel s_{12}\end{aligned}$$

Θα πρέπει να ελέγξετε ξανά ότι, δεδομένης αυτής της κοινής χρήσης, όλα τα κατάλληλα σύνολα, και μόνο αυτά, μπορούν να ανακτήσουν το μυστικό. Βλέπουμε ότι σε αυτή την περίπτωση κάθε μερίδιο περιέχει τρεις φορές περισσότερα δεδομένα από το υποκείμενο μυστικό.

6.6.2 Επαναλαμβανόμενη Κοινή Χρήση Μυστικών

Το παραπάνω σχήμα δεν είναι το μόνο για γενική δομή πρόσβασης. Ένα άλλο σχήμα ονομάζεται επαναλαμβανόμενη κοινή χρήση μυστικών (replicated secret sharing). Σε αυτό το σχήμα, δημιουργούνται πρώτα τα σύνολα όλων των μέγιστων συνόλων που δεν πληρούν τις προϋποθέσεις. Αυτά είναι τα σύνολα όλων των συμμετεχουσών οντότητων, έτσι ώστε, εάν προσθέστε μια μόνο νέα οντότητα σε κάθε σύνολο θα αποκτήσετε ένα κατάλληλο σύνολο. Αν ονομάσουμε αυτά τα σύνολα A_1, \dots, A_t , σχηματίζουμε τα συμπληρώματα τους, δηλ. $B_i = P/A_i$. Στη συνέχεια, για κάθε σύνολο B_i δημιουργείται ένα σύνολο από μερίδια s_i , έτσι ώστε:

$$s = s_1 \oplus \dots \oplus s_t$$

Σε κάθε οντότητα δίνεται το μερίδιο s_i , εάν αυτό περιλαμβάνεται στο σύνολο B_i .

Παράδειγμα 1: Τα σύνολα των μέγιστων συνόλων που δεν πληρούν τις προϋποθέσεις για αυτό το παράδειγμα είναι:

$$A_1 = \{\Pi, A, \Upsilon\}, A_2 = \{A, \Sigma\} \text{ και } A_3 = \{\Upsilon, \Sigma\}$$

Σχηματίζοντας τα συμπληρώματά τους παίρνουμε τα σύνολα:

$$B_1 = \{\Sigma\}, B_2 = \{\Pi, \Upsilon\} \text{ και } A_3 = \{\Pi, A\}$$

Δημιουργούμε τρία μερίδια s_1, s_2 και s_3 , τέτοια ώστε $s = s_1 \oplus s_2 \oplus s_3$, και στη συνέχεια ορίζουμε τα μερίδια ως:

$$\begin{aligned} s_\Pi &= s_2 \parallel s_3 \\ s_A &= s_3 \\ s_\Upsilon &= s_2 \\ s_\Sigma &= s_1 \end{aligned}$$

Και πάλι μπορούμε να ελέγξουμε ότι μόνο τα κατάλληλα σύνολα μπορούν να ανακτήσουν το μυστικό.

Παράδειγμα 2: Για τη δομή πρόσβασης κατωφλίου 2-από-4 λαμβάνουμε τα ακόλουθα μέγιστα σύνολα που δεν πληρούν τις προϋποθέσεις: ως:

$$A_1 = \{A\}, A_2 = \{B\}, A_3 = \{\Gamma\} \text{ και } A_4 = \Delta$$

Σχηματίζοντας τα συμπληρώματά τους παίρνουμε:

$$B_1 = \{B, \Gamma, \Delta\}, B_2 = \{A, \Gamma, \Delta\}, B_3 = \{A, B, \Delta\} \text{ και } B_4 = \Delta$$

Σχηματίζουμε τα τέσσερα μερίδια τέτοια ώστε $s = s_1 \oplus s_2 \oplus s_3 \oplus s_4$ και:

$$\begin{aligned} s_A &= s_2 \parallel s_3 \parallel s_4 \\ s_B &= s_1 \parallel s_3 \parallel s_4 \\ s_\Gamma &= s_1 \parallel s_2 \parallel s_4 \\ s_\Delta &= s_1 \parallel s_2 \parallel s_3 \end{aligned}$$

Ενώ τα παραπάνω σχήματα παρέχουν έναν μηχανισμό για την δημιουργία ενός σχήματος μυστικών κοινής χρήσης για οποιαδήποτε δομή μονότονης πρόσβασης, εντούτοις στην πράξη έχουν αποδειχθεί ότι είναι πολύ αναποτελεσματικά. Ειδικά για τη δομή πρόσβασης κατωφλίου είναι ιδιαίτερα κακές, κυρίως καθώς αυξάνεται ο αριθμός των συμμετεχουσών οντοτήτων.

6.7 Μονάδες Ασφαλείας Υλισμικού

Οι μονάδες ασφαλείας υλισμικού (Hardware Security Module – HSM) είναι κρυπτογραφικές μονάδες που παρέχουν υψηλά επίπεδα προστασίας ευαίσθητων δεδομένων. Πρόκειται για φυσικές συσκευές υλισμικού που χρησιμοποιούνται για την διαχείριση κρυπτογραφικών κλειδιών για κρίσιμες λειτουργίες όπως κρυπτογράφηση, αποκρυπτογράφηση, ηλεκτρονικές υπογραφές (βλέπε Ενότητα 7.2.1.4), και αυθεντικοποίηση για κάθε είδους χρήση, συμπεριλαμβανομένων και όλων των εφαρμογών.

Αυτές οι συσκευές μπορεί να είναι ανεξάρτητες συσκευές, κάρτες υλισμικού, ή να είναι ενσωματωμένες σε άλλο υλισμικό. Μπορούν να συνδεθούν σε διακομιστή δικτύου ή να χρησιμοποιηθούν ως αυτόνομη συσκευή εκτός σύνδεσης.

Οι οργανισμοί και οι επιχειρήσεις χρησιμοποιούν HSM για τις ανάγκες των κρυπτογραφικών λειτουργιών που σχετίζονται με συναλλαγές, ταυτότητες και εφαρμογές, και για να ελέγχουν την πρόσβαση σε αυτές τις λειτουργίες. Για παράδειγμα, μια εταιρεία μπορεί να χρησιμοποιήσει ένα HSM για την προστασία εμπορικών μυστικών ή πνευματικής ιδιοκτησίας διασφαλίζοντας ότι μόνο εξουσιοδοτημένα άτομα μπορούν να έχουν πρόσβαση στο HSM για να ολοκληρώσουν μια μεταφορά κλειδιού κρυπτογράφησης.

Τα ακόλουθα χαρακτηριστικά συμβάλλουν στην ασφάλεια ενός HSM:

- **Ασφαλής σχεδιασμός.** Τα HSM χρησιμοποιούν ειδικά σχεδιασμένο υλικό που συμμορφώνεται και με κυβερνητικά πρότυπα, όπως το πρότυπο FIPS 140-3 [4], τα Common Criteria³ και οι απαιτήσεις HSM του Payment Card Industry (PCI).
- **Ανθεκτικό στις παραβιάσεις (tamper resistant).** Τα HSM υποβάλλονται σε διαδικασία security hardening⁴ για να γίνουν ανθεκτικά σε παραβιάσεις αλλά και ακούσιες ζημιές. Έχουν σχεδιαστεί για να κάνουν εμφανή τα σημάδια παραβιάσης. Ορισμένα HSMs καθίστανται ανενεργά ή διαγράφονται αποθηκευμένα σε αυτά κρυπτογραφικά κλειδιά, εάν εντοπιστεί παραβιάση.
- **Ασφαλές λειτουργικό σύστημα.** Διαθέτουν λειτουργικό σύστημα εστιασμένο στην ασφάλεια.
- **Απομόνωση.** Τα HSMs εγκαθίστανται σε μια ασφαλή φυσική περιοχή του υπολογιστικού κέντρου, είτε εντός οργανισμού είτε σε τρίτους, παρόχους υπηρεσιών, για την αποτροπή μη εξουσιοδοτημένης φυσικής πρόσβασης.
- **Έλεγχοι πρόσβασης.** Τα HSMs ελέγχουν την πρόσβαση στα δεδομένα που προστατεύουν.

6.8 Συστήματα Διαχείρισης Κλειδιών

Σε πολλούς μεγάλους οργανισμούς υπάρχει ανάγκη να συστηματοποιηθεί ο κύκλος ζωής των κρυπτογραφικών κλειδιών. Αυτό γίνεται συνήθως χρησιμοποιώντας ένα Σύστημα Διαχείρισης Κρυπτογραφικών Κλειδιών (ΣΔΚΚ), το οποίο αποτελεί ένα αυτοματοποιημένο σύστημα που περιλαμβάνει στοιχεία υλικού και λογισμικού που εφαρμόζουν την απαιτούμενη πολιτική για τη διαχείριση των κλειδιών. Για παράδειγμα, εάν τα κλειδιά διατηρούνται σε ασφαλείς κρυπτογραφικές μονάδες υλικού, τότε είναι κοινή πρακτική να ενεργοποιείται μόνο η εξαγωγή κλειδιών από τις μονάδες υλικού με κάποια μορφή κρυπτογράφησης του ίδιου του κλειδιού με κάποιο άλλο κλειδί (περιτύλιξη κλειδιού – key wrapping). Ένα σύστημα διαχείρισης κρυπτογραφικών κλειδιών διασφαλίζει ότι μια τέτοια πολιτική επιβάλλεται, χωρίς οι χρήστες να μπορούν να την παρακάμψουν.

Το πρότυπο NIST 800-130 [18] ορίζει ένα πλαίσιο για τον σχεδιασμό ΣΔΚΚ. Παρέχει μια περιγραφή των απαιτήσεων που πρέπει να λαμβάνονται υπόψη κατά το σχεδιασμό ενός ΣΔΚΚ, καθώς και της απαιτούμενης τεκμηρίωσης.

³Τα “Common Criteria for Information Technology Security Evaluation”, γνωστά ως Common Criteria ή CC, αποτελούν ένα διεθνές πρότυπο (ISO/IEC 15408) για την πιστοποίηση ασφάλειας υπολογιστών. Αποτελούν ένα πλαίσιο στο οποίο οι χρήστες συστημάτων υπολογιστών μπορούν να προσδιορίσουν τις λειτουργικές απαιτήσεις και τις απαιτήσεις ασφαλείας (Security Functional Requirements – SFR και Security Assurance Requirements – SAR, αντίστοιχα) σε έναν στόχο ασφαλείας (Security Target - ST) και μπορούν να ληφθούν από Προφίλ προστασίας (Protection Profiles – PP). Οι κατασκευαστές μπορούν στη συνέχεια να εφαρμόσουν αυτά τα χαρακτηριστικά ασφαλείας στα προϊόντα τους ενώ τα εργαστήρια δοκιμών μπορούν να αξιολογήσουν τα προϊόντα για να προσδιορίσουν εάν καλύπτονται πραγματικά οι απαιτήσεις. Με άλλα λόγια, το Common Criteria παρέχει διαβεβαίωση ότι η διαδικασία καθορισμού προδιαγραφών, υλοποίησης και αξιολόγησης ενός προϊόντος ασφαλείας υπολογιστών έχει διεξαχθεί με αυστηρό και τυπικό τρόπο, σε επίπεδο ανάλογο με το περιβάλλον-στόχο για χρήση. Το Common Criteria διατηρεί μια λίστα πιστοποιημένων προϊόντων, συμπεριλαμβανομένων των λειτουργικών συστημάτων, συστημάτων ελέγχου πρόσβασης, βάσεων δεδομένων και συστημάτων διαχείρισης κλειδιών.

⁴Η διαδικασία κατά την οποία υπηρεσίες και συστήματα καθίστανται λιγότερο επιρρεπή σε επιθέσεις, εφαρμόζοντας την αρχή των ελάχιστων προνομίων, ελαχιστοποιώντας την πρόσβαση σε πόρους, απενεργοποιώντας τις θύρες και τα πρωτόκολλα και γενικά μειώνοντας τη δυνατότητα εκμετάλλευσης ευπαθειών και τον αριθμό αυτών στις υπηρεσίες και στα συστήματα.

Σύμφωνα με το πρότυπο, ένα ΣΔΚΚ αποτελείται από πολιτικές, διαδικασίες, στοιχεία και συσκευές που χρησιμοποιούνται για την προστασία, τη διαχείριση και το διαμοιρασμό κρυπτογραφικών κλειδιών και ορισμένων ειδικών πληροφοριών, που ονομάζονται (συσχετισμένα) μεταδεδομένα. Ένα ΣΔΚΚ περιλαμβάνει όλες τις συσκευές ή τα υποσυστήματα που μπορούν να έχουν πρόσβαση σε ένα μη κρυπτογραφημένο κλειδί ή στα μεταδεδομένα του. Τα κρυπτογραφημένα κλειδιά και τα κρυπτογραφικά προστατευμένα μεταδεδομένα τους μπορούν να γίνονται αντικείμενα επεξεργασίας από υπολογιστές, να μεταδίδονται μέσω συστημάτων επικοινωνιών και να αποθηκεύονται σε μέσα που δεν θεωρούνται μέρος ενός ΣΔΚΚ.

Βιβλιογραφία

- [1] Elaine Barker. *Recommendation for key management:: part 1 - general*. Tech. rep. NIST SP 800-57pt1r5. Gaithersburg, MD: National Institute of Standards and Technology, May 2020, NIST SP 800-57pt1r5. DOI: 10.6028/NIST.SP.800-57pt1r5.
- [2] Elaine Barker and William C Barker. *Recommendation for key management:: part 2 – best practices for key management organizations*. Tech. rep. NIST SP 800-57pt2r1. Gaithersburg, MD: National Institute of Standards and Technology, May 2019, NIST SP 800-57pt2r1. DOI: 10.6028/NIST.SP.800-57pt2r1.
- [3] *Information technology – Security techniques – Security requirements for cryptographic modules*. Standard. International Organization for Standardization/International Electrotechnical Commission, Aug. 2012, pp. 1–72.
- [4] National Institute of Standards and Technology. *Security requirements for cryptographic modules*. Tech. rep. NIST FIPS 140-3. Gaithersburg, MD: National Institute of Standards and Technology, Apr. 2019, NIST FIPS 140-3. DOI: 10.6028/NIST.FIPS.140-3.
- [5] *Information technology – IT Security techniques – Key management – Part 2: Mechanisms using symmetric techniques*. Standard. International Organization for Standardization/International Electrotechnical Commission, Oct. 2018, pp. 1–28.
- [6] *Financial services – Key management (retail)*. Standard. International Organization for Standardization/International Electrotechnical Commission, Feb. 2023, pp. 1–115.
- [7] Elaine Barker, Allen Roginsky, and Richard Davis. *Recommendation for cryptographic key generation*. Tech. rep. NIST SP 800-133r2. Edition: r2. Gaithersburg, MD: National Institute of Standards and Technology, June 2020, NIST SP 800-133r2. DOI: 10.6028/NIST.SP.800-133r2.
- [8] Lily Chen. *Recommendation for Key Derivation Using Pseudorandom Functions*. Tech. rep. NIST SP 800-108r1. Gaithersburg, MD: National Institute of Standards and Technology, 2022, NIST SP 800-108r1. DOI: 10.6028/NIST.SP.800-108r1.
- [9] Elaine Barker, Lily Chen, Allen Roginsky, Apostol Vassilev, and Richard Davis. *Recommendation for pair-wise key-establishment schemes using discrete logarithm cryptography*. Tech. rep. NIST SP 800-56Ar3. Gaithersburg, MD: National Institute of Standards and Technology, Apr. 2018, NIST SP 800-56Ar3. DOI: 10.6028/NIST.SP.800-56Ar3.
- [10] Elaine Barker et al. *Recommendation for pair-wise key establishment using integer factorization cryptography*. Tech. rep. NIST SP 800-56Br2. Gaithersburg, MD: National Institute of Standards and Technology, Mar. 2019, NIST SP 800-56Br2. DOI: 10.6028/NIST.SP.800-56Br2.
- [11] Colin Boyd and Anish Mathuria. *Protocols for Authentication and Key Establishment*. Ed. by Ueli Maurer et al. Information Security and Cryptography. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003. ISBN: 978-3-642-07716-6 978-3-662-09527-0. DOI: 10.1007/978-3-662-09527-0.

- [12] Whitfield Diffie and Martin E. Hellman. "New Directions in Cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. doi: 10.1109/TIT.1976.1055638.
- [13] Eric Rescorla. *Diffie-Hellman Key Agreement Method*. RFC 2631. June 1999. doi: 10.17487/RFC2631.
- [14] B.C. Neuman and T. Ts'o. "Kerberos: an authentication service for computer networks". In: *IEEE Communications Magazine* 32.9 (Sept. 1994), pp. 33–38. issn: 0163-6804. doi: 10.1109/35.312841.
- [15] L. Zhu and B. Tung. *Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)*. en. Tech. rep. RFC4556. RFC Editor, June 2006, RFC4556. doi: 10.17487/rfc4556.
- [16] S. Josefsson. *Using Kerberos Version 5 over the Transport Layer Security (TLS) Protocol*. en. Tech. rep. RFC6251. RFC Editor, May 2011, RFC6251. doi: 10.17487/rfc6251.
- [17] Roger M. Needham and Michael D. Schroeder. "Using encryption for authentication in large networks of computers". en. In: *Communications of the ACM* 21.12 (Dec. 1978), pp. 993–999. issn: 0001-0782, 1557-7317. doi: 10.1145/359657.359659.
- [18] Elaine Barker, Miles Smid, Dennis Branstad, and Santosh Chokhani. *A Framework for Designing Cryptographic Key Management Systems*. Tech. rep. NIST SP 800-130. National Institute of Standards and Technology, Aug. 2013, NIST SP 800–130. doi: 10.6028/NIST.SP.800-130.

ΚΕΦΑΛΑΙΟ 7

ΥΠΟΔΟΜΕΣ ΔΗΜΟΣΙΟΥ ΚΛΕΙΔΙΟΥ ΚΑΙ ΥΠΗΡΕΣΙΕΣ ΕΜΠΙΣΤΟΣΥΝΗΣ

Περίληψη

Η χρήση κρυπτογραφίας δημοσίου κλειδιού απαιτεί τη δημιουργία και λειτουργία υποδομών δημοσίου κλειδιού (Ενότητα 7.1) που θέτουν το πλαίσιο και εξασφαλίζουν τις προϋποθέσεις για την ασφαλή και αποτελεσματική διαχείριση κλειδιών και ψηφιακών πιστοποιητικών. Στο κεφάλαιο αυτό αναλύονται τα θέματα που αφορούν στη διαχείριση των κλειδιών ασύμμετρης κρυπτογραφίας και ψηφιακών πιστοποιητικών σύμφωνα με το πρότυπο X.509 (Ενότητα 7.1.2.2). Επιπλέον αναλύονται θέματα που αφορούν στο νομοθετικό πλαίσιο που διέπει τα είδη των ηλεκτρονικών υπογραφών (Ενότητα 7.2.1) και την παροχή υπηρεσιών εμπιστοσύνης, όπως είναι αυτές της χρονοσήμανσης (Ενότητα 7.2.4) και της συστημένης παράδοσης (Ενότητα 7.2.3). Τέλος, στην Ενότητα 7.3 παρουσιάζονται θέματα που αφορούν στα ψηφιακά πορτοφόλια.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών εννοιών της Κρυπτογραφίας που παρατίθενται στα εισαγωγικά κεφάλαια (Κεφάλαιο 1 έως 3) αυτού του βιβλίου.

7.1 Υποδομές Δημοσίου Κλειδιού

Η ασύμμετρη κρυπτογραφία αποτελεί ακρογωνιαίο λίθο της ασφαλούς μετάδοσης δεδομένων και του ελέγχου ταυτότητας, ενώ βασίζεται στη χρήση ζευγών δημόσιων και ιδιωτικών κλειδιών. Ωστόσο, η αποτελεσματική εφαρμογή της ασύμμετρης κρυπτογραφίας εξαρτάται από ένα κρίσιμο στοιχείο: την Υποδομή Δημόσιου Κλειδιού – ΥΔΚ (Public Key Infrastructure). Η ΥΔΚ είναι ένα ολοκληρωμένο σύστημα που διευκολύνει τη δημιουργία, τη διανομή, τη διαχείριση και την ανάκληση ψηφιακών πιστοποιητικών και κρυπτογραφικών κλειδιών. Χωρίς μια ορθά υλοποιημένη και ασφαλή ΥΔΚ, η δυναμική της ασύμμετρης κρυπτογραφίας παραμένει αναξιοποίητη, καθώς η θεμελιώδης πρόκληση της ασφαλούς μετάδοσης των δημόσιων κλειδιών και της επαλήθευσης της αυθεντικότητας των οντοτήτων σε ένα περιβάλλον ανοιχτού δικτύου δεν μπορεί να αντιμετωπιστεί επαρκώς. Επομένως, η κατανόηση του κεντρικού ρόλου μιας ΥΔΚ είναι απαραίτητη για την κατανόηση της συμβολής της σε ένα ασφαλές κρυπτοσύστημα που βασίζεται στη χρήση κρυπτογραφίας

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

δημοσίου κλειδιού.

Μία ΥΔΚ είναι ένας συνδυασμός στοιχείων υλισμικού/λογισμικού, πολιτικών και διαδικασιών που συνολικά στοχεύουν στην ορθή διαχείριση των κλειδιών ασύμμετρης κρυπτογραφίας. Βασίζεται και υποστηρίζει την ορθή και ασφαλή χρήση ψηφιακών πιστοποιητικών που χρησιμοποιούνται για την ψηφιακή ταυτοποίηση των κατόχων τους και την διακίνηση αυθεντικοποιημένων αντιγράφων δημοσίων κλειδιών. Με αυτόν τον τρόπο, τα πιστοποιητικά συσχετίζουν τους χρήστες με τα δημόσια κλειδιά τους.

Μια ΥΔΚ αποτελείται από τα ακόλουθα στοιχεία και οντότητες:

- Πάροχοι Υπηρεσιών Εμπιστοσύνης – ΠΥΕ (Trust Service Provider) γνωστοί και ως Αρχές Πιστοποίησης – ΑΠ (Certification Authorities): Αποτελούν τις έμπιστες τρίτες οντότητες που εκδίδουν και διαχειρίζονται τα ψηφιακά πιστοποιητικά των τελικών χρηστών – κατόχων ψηφιακών πιστοποιητικών. Οι ΠΥΕ πιστοποιούν την εγκυρότητα των ζευγών «δημόσιο κλειδί – κάτοχος» μέσω ενός πιστοποιητικού δημόσιου κλειδιού.
- Πολιτική Πιστοποίησης – ΠΠ (Certification Policy): Ορίζεται από το πρότυπο X.509 [1] ως «ένα σύνολο κανόνων που υποδεικνύει τη δυνατότητα εφαρμογής του πιστοποιητικού σε μια συγκεκριμένη κοινότητα ή/και εφαρμογές με κοινές απαιτήσεις ασφαλείας».

Μια ΠΠ καθοδηγεί τα Βασιζόμενα Μέρη (Relying Parties)¹, ώστε να αξιολογούν κατά πόσο ένα ψηφιακό πιστοποιητικό που χρησιμοποιούν είναι κατάλληλο για χρήση σε μια συγκεκριμένη εφαρμογή. Επιπλέον παρέχει προστασία στον ΠΥΕ που την υιοθετεί, δηλώνοντας το προβλεπόμενο εύρος χρήσεων για τα πιστοποιητικά που εκδίδει.

- Δήλωση Πρακτικής Πιστοποίησης – ΔΠΠ (Certification Practice Statement): Ορίζεται στο RFC 3647 [2], ως «μια δήλωση των πρακτικών που χρησιμοποιεί μια αρχή πιστοποίησης κατά την έκδοση πιστοποιητικών».

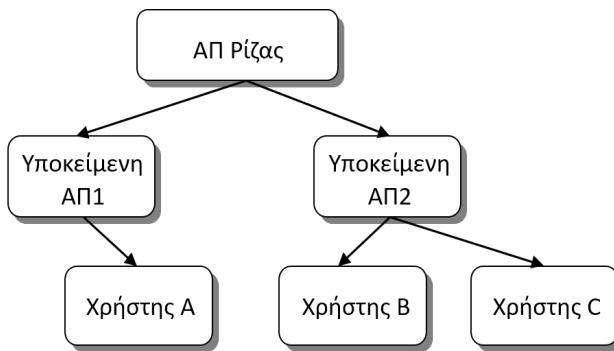
Η ΔΠΠ είναι ένα έγγραφο που αφορά συγκεκριμένο ΠΥΕ, ενώ μια ΠΠ μπορεί να είναι κοινή σε πολλούς ΠΥΕ. Μια ΔΠΠ τεκμηριώνει επίσης τα μέσα με τα οποία οι τελικοί χρήστες και τα βασιζόμενα μέρη αλληλεπιδρούν με έναν ΠΥΕ.

- Αρχή Εγγραφής – ΑΕ (Registration Authority): Αποτελεί τη διεπαφή μεταξύ των χρηστών και του ΠΥΕ. Καταχωρεί και επαληθεύει την ταυτότητα των χρηστών και προωθεί αιτήματα σχετικά με την έκδοση και διαχείριση πιστοποιητικών στον ΠΥΕ.
- Σύστημα Διανομής Πιστοποιητικών – ΣΔΠ: Αφορά τους μηχανισμούς που χρησιμοποιούνται για την διανομή των πιστοποιητικών στους τελικούς χρήστες και τα βασιζόμενα μέρη, για παράδειγμα, διανομή από τους ίδιους τους χρήστες ή με τη χρήση διακομιστή καταλόγου όπως το LDAP (Lightweight Directory Access Protocol).
- Εφαρμογές ΥΔΚ: Όπως, για παράδειγμα, η ηλεκτρονική αλληλογραφία (email) και η επικοινωνία μεταξύ του εξυπηρετητή (server) και του περιηγητή (browser) ιστού.

7.1.1 Μοντέλα ΥΔΚ

Το πιο συχνά χρησιμοποιούμενο μοντέλο ΥΔΚ είναι το ιεραρχικό μοντέλο που απεικονίζεται στο Σχήμα 7.1. Σε αυτό το μοντέλο η Αρχή Πιστοποίησης Ρίζας (Root Certification Authority) αποτελεί ένα από τα κεντρικά στοιχεία της ΥΔΚ καθώς αποτελεί την αρχή που βρίσκεται στο ανώτατο επίπεδο της ιεραρχίας των πιστοποιητικών και έχει έναν κρίσιμο ρόλο στην ασφαλή λειτουργία μιας ΥΔΚ.

¹Το βασιζόμενο μέρος (relying party) είναι φυσικό ή νομικό πρόσωπο το οποίο βασίζεται σε ηλεκτρονική ταυτοποίηση ή σε υπηρεσία εμπιστοσύνης.



Σχήμα 7.1: Ιεραρχικό μοντέλο ΥΔΚ.

Η Αρχή Πιστοποίησης Ρίζας (ΑΠ Ρίζας) εκδίδει το Πιστοποιητικό Ρίζας, το οποίο αποτελεί το ανώτατο επίπεδο της ιεραρχίας των πιστοποιητικών σε μια ΥΔΚ. Αυτό το πιστοποιητικό είναι υπεύθυνο για την αυθεντικοποίηση της ίδιας της ΑΠ Ρίζας, δημιουργώντας ένα αξιόπιστο σημείο εκκίνησης και τις απαιτούμενες σχέσεις εμπιστοσύνης για τον έλεγχο της εγκυρότητας των υπολοίπων πιστοποιητικών στην ιεραρχία. Πρόκειται για ένα αυτο-υπογεγραμμένο πιστοποιητικό (μερικές φορές ονομάζεται επίσης και «άγκυρα εμπιστοσύνης» (trust anchor)) καθώς εκδίδεται από την ίδια την ΑΠ Ρίζας, η οποία πιστοποιεί τη σύνδεση του δημοσίου κλειδιού της με την ταυτότητά της.

Τα πιστοποιητικά των ΑΠ Ρίζας διαδραματίζουν βασικό ρόλο σε πολλά πρωτόκολλα και εφαρμογές και γενικά διατηρούνται σε αυτό που συχνά αποκαλείται χώρος αποθήκευσης πιστοποιητικών ρίζας. Ένα σημαντικό κομμάτι της σωστής διαμόρφωσης εφαρμογών και πρωτοκόλλων συνίσταται στη διασφάλιση ότι μόνο τα κατάλληλα πιστοποιητικά ρίζας φορτώνονται στον χώρο αποθήκευσης πιστοποιητικών ρίζας. Στα περισσότερα λειτουργικά συστήματα, όπως Microsoft Windows και Unix-like συστήματα, υπάρχουν χώροι αποθήκευσης πιστοποιητικών ρίζας που διατηρούνται από το λειτουργικό σύστημα και είναι διαθέσιμα σε πρωτόκολλα και εφαρμογές που θα επιλέξουν να τα χρησιμοποιήσουν. Παρόμοια λειτουργικότητα υπάρχει και στα λειτουργικά συστήματα της Apple, η οποία είναι γνωστή ως “Keychain”. Ορισμένες εφαρμογές, που προορίζονται να είναι φορητές μεταξύ λειτουργικών συστημάτων, μπορούν να διατηρούν τους δικούς τους χώρους αποθήκευσης πιστοποιητικών ρίζας και έχουν επίσης μια δυνατότητα που τους επιτρέπει να μοιράζονται ένα χώρο αποθήκευσης πιστοποιητικών ρίζας με άλλες εφαρμογές. Εάν ένα ψηφιακό πιστοποιητικό ρίζας δε γίνεται αποδεκτό ως αξιόπιστο από μια εφαρμογή, τότε οποιοδήποτε πιστοποιητικό που εκδίδεται κάτω από την ιεραρχία της ΑΠ Ρίζας δε θα γίνεται αποδεκτό ως αξιόπιστο, εκτός και αν ο χρήστης δηλώσει ρητά την εμπιστοσύνη του σε αυτό το πιστοποιητικό.

Τα πιστοποιητικά που βρίσκονται κάτω από την ιεραρχία οποιασδήποτε από τις αξιόπιστες ΑΠ Ρίζας, όπως είναι αυτά των Υποκείμενων Αρχών Πιστοποίησης, κληρονομούν τη σχέση εμπιστοσύνης που υπάρχει προς την ΑΠ Ρίζας και γίνονται αποδεκτά ως αξιόπιστα από τις εφαρμογές που χρησιμοποιούν τα πιστοποιητικά τους. Για παράδειγμα, τα προγράμματα περιήγησης Διαδικτύου κάνουν χρήση αυτών των αξιόπιστων ΑΠ για τη δημιουργία ασφαλών καναλιών επικοινωνίας στο πλαίσιο της χρήσης του TLS (Transport Layer Security) πρωτοκόλλου (βλέπε Ενότητα 10.2).

Μια Υποκείμενη Αρχή Πιστοποίησης (ΥΑΠ) είναι μια ενδιάμεση αρχή που βρίσκεται ανάμεσα στην ΑΠ Ρίζας και τους τελικούς χρήστες. Οι ΥΑΠ, μπορούν να πιστοποιούν άλλες υποκείμενες αρχές, δημιουργώντας μια ιεραρχία που επιτρέπει την ενισχυμένη ασφάλεια και διαχείριση των πιστοποιητικών.

Το ψηφιακό πιστοποιητικό μιας ΥΑΠ, η οποία μπορεί να αποτελεί και την Εκδόσα ΑΠ (Issuing Certification Authority), δηλαδή την ΑΠ που εκδίδει ψηφιακά πιστοποιητικά για τους τελικούς χρήστες, εκδίδεται από την την άμεσα υπερκείμενη στην ιεραρχία ΑΠ ή από την ΑΠ Ρίζας. Στη συνέχεια, το πιστοποιητικό μιας Εκδόσας ΑΠ χρησιμοποιείται για την έκδοση πιστοποιητικών σε τελικούς χρήστες για διάφορους σκοπούς. Μερικοί από τους σκοπούς αφορούν πιστοποιητικά υπογραφής εγγράφων, πιστοποιητικά TLS για τη δημιουργία ασφαλών συνδέσεων στο διαδίκτυο, ασφαλούς ηλεκτρονικού ταχυδρομείου S/MIME, και υπο-

γραφής κώδικα.

7.1.2 Ψηφιακά Πιστοποιητικά

Τα Ψηφιακά Πιστοποιητικά (Digital Certificates) αποτελούν ψηφιακά αρχεία που πιστοποιούν την συσχέτιση μιας συγκεκριμένης ταυτότητας, αυτής του χρήστη και κατόχου του πιστοποιητικού, με ένα δημόσιο κλειδί το οποίο μπορεί να χρησιμοποιείται για την επαλήθευση μιας ψηφιακής υπογραφής του κατόχου ή για την κρυπτογράφηση δεδομένων που αποστέλλονται προς τον κάτοχο του πιστοποιητικού. Το πιστοποιητικό χρησιμοποιείται για τη διανομή του δημόσιου κλειδιού του χρήστη σε άλλα ενδιαφερόμενα μέρη, γνωστά ως βασιζόμενα μέρη (relying parties), καθώς βασίζονται στις διαβεβαιώσεις που παρέχονται από τον ΠΥΕ και την πολιτική έκδοσης πιστοποιητικού για να κάνουν χρήση του σχετικού δημόσιου κλειδιού.

Οι εγγυήσεις για την συσχέτιση του κατόχου του πιστοποιητικού με το δημόσιο κλειδί παρέχονται από την Εκδόσα ΑΠ μέσω της ψηφιακής της υπογραφής για το σύνολο του πιστοποιητικού. Η Εκδόσα ΑΠ είναι ένας αξιόπιστος ΠΥΕ που δημιουργεί και υπογράφει το πιστοποιητικό αφού επαληθεύσει την ταυτότητα του χρήστη και την κατοχή του δημοσίου κλειδιού από τον χρήστη.

Οι πληροφορίες που απαιτούνται για την αναγνώριση του κατόχου του δημόσιου κλειδιού καταχωρούνται στο πιστοποιητικό. Σε πολλές περιπτώσεις, η ΑΠ θα μεταβιβάσει την ευθύνη για την επαλήθευση της ταυτότητας του υποκειμένου κατά την εγγραφή του σε μια Αρχή Εγγραφής (AE). Μόλις επιβεβαιωθεί η ταυτότητα από την AE, το πιστοποιητικό υπογράφεται (ψηφιακά) από την ΑΠ με το δικό της ιδιωτικό κλειδί, διασφαλίζοντας έτσι την εγκυρότητα των πληροφοριών που αναγράφονται σε αυτό.

Τα πιστοποιητικά μπορούν να εκδοθούν για άτομα, εταιρείες, λογισμικό ή διακομιστές. Αντίστοιχα, μπορούμε να μιλάμε για προσωπικά πιστοποιητικά, εταιρικά πιστοποιητικά ή πιστοποιητικά λογισμικού ή διακομιστή. Ανάλογα με τις επιμέρους απαιτήσεις, μπορούν να χρησιμοποιηθούν διαφορετικές μέθοδοι για την επαλήθευση μιας ταυτότητας. Οι διαδικασίες επαλήθευσης ταυτότητας ποικίλλουν από μια απλή επαλήθευση της διεύθυνσης ήλεκτρονικού ταχυδρομείου ενός ατόμου έως περιπτώσεις όπου ένα άτομο πρέπει να παρουσιάσει προσωπικά την ταυτότητά του για να λάβει πιστοποιητικό. Αυτό και, ευρύτερα, οι διαδικασίες που χρησιμοποιούνται από τον ΠΥΕ για την έκδοση και διαχείριση των πιστοποιητικών, επηρεάζουν το είδος του πιστοποιητικού καθώς και το επίπεδο εμπιστοσύνης που μπορεί να έχει κάποιο βασιζόμενο μέρος σε αυτό.

7.1.2.1 Αναγκαιότητα Χρήσης Ψηφιακών Πιστοποιητικών

Τα προβλήματα που σχετίζονται με τη διανομή κλειδιών κατά τη χρήση ασύμμετρων κρυπτοσυστημάτων συχνά υποτιμώνται ή δεν αναγνωρίζονται πραγματικά. Με τα δημόσια κλειδιά να ανταλλάσσονται εν γνώσει τους μέσω ενός μη ασφαλούς καναλιού, πολλοί άνθρωποι φαίνεται να πιστεύουν ότι η «υποκλοπή» αυτών των κλειδιών είναι η μόνη απειλή για την ασφάλεια της επικοινωνίας.

Ωστόσο, πρέπει να ληφθούν επιπλέον υπόψη ένα σύνολο πιθανών ενεργών επιθέσεων όπως είναι η ακόλουθη:

- Το δημόσιο κλειδί P_A του χρήστη A αντικαθίσταται κατά την αποστολή του στον ΠΥΕ με άλλο κλειδί $P_{A'}$, το οποίο ανήκει σε έναν κακόβουλο χρήστη M.
- Στη βάση δεδομένων καταχώρησης κλειδιών τελικών χρηστών, για τον χρήστη A καταχωρείται το «ψεύτικο» κλειδί $P_{A'}$. Ένας τρίτος χρήστης B χρησιμοποιεί αυτό το κλειδί για να κρυπτογραφήσει ένα μήνυμα που, θεωρητικά, απευθύνεται στον A. Στην πράξη ωστόσο, αυτό το μήνυμα κρυπτογραφείται για τον κακόβουλο χρήστη M, στον οποίο ανήκει το κλειδί $P_{A'}$.
- Ο χρήστης M υποκλέπτει το μήνυμα και το αποκρυπτογραφεί χρησιμοποιώντας το ιδιωτικό κλειδί που σχετίζεται με το δημόσιο κλειδί $P_{A'}$. Έτσι αποκτά πρόσβαση στο αποκρυπτογραφημένο μήνυμα.
- Στη συνέχεια, ο χρήστης M μπορεί να κρυπτογραφήσει εκ νέου το μήνυμα με το δημόσιο κλειδί του χρήστη A και να το στείλει στον χρήστη A, ο οποίος το αποκρυπτογραφεί με το δικό του ιδιωτικό

κλειδί.

Ούτε ο χρήστης Α ούτε ο Β είναι πιθανό να έχουν παρατηρήσει την επίθεση. Τα πιστοποιητικά χρησιμοποιούνται για την προστασία από επίθεση αυτού του είδους, καθώς επιτρέπουν την ασφαλή ανταλλαγή αυθεντικοποιημένων αντιγράφων των δημοσίων κλειδιών των χρηστών, προστατεύοντάς τα από επιθέσεις πλαστοπροσωπίας.

7.1.2.2 Πιστοποιητικά X.509 v3

Οι οργανισμοί CCITT και ISO έχουν συμφωνήσει για μια τυποποίηση των ψηφιακών πιστοποιητικών με το πρωτόκολλο X.509. Επί του παρόντος, στην τελευταία του έκδοση 3 [3, 1], το πρωτόκολλο X.509 περιγράφει τη δομή των πιστοποιητικών, η οποία έχει ήδη υιοθετηθεί από διάφορα άλλα πρότυπα και αποτελεί το πιο ευρέως χρησιμοποιούμενο πρότυπο πιστοποιητικών.

Σύμφωνα με το πρότυπο X.509 v3, ένα πιστοποιητικό πρέπει να περιέχει τουλάχιστον πληροφορίες για τον κάτοχο του πιστοποιητικού, το δημόσιο κλειδί του κατόχου, καθώς και την ψηφιακή υπογραφή του ΠΥΕ. Έτσι, ένα πιστοποιητικό X.509 περιλαμβάνει ένα σύνολο πεδίων που αφορούν στις ακόλουθες πληροφορίες:

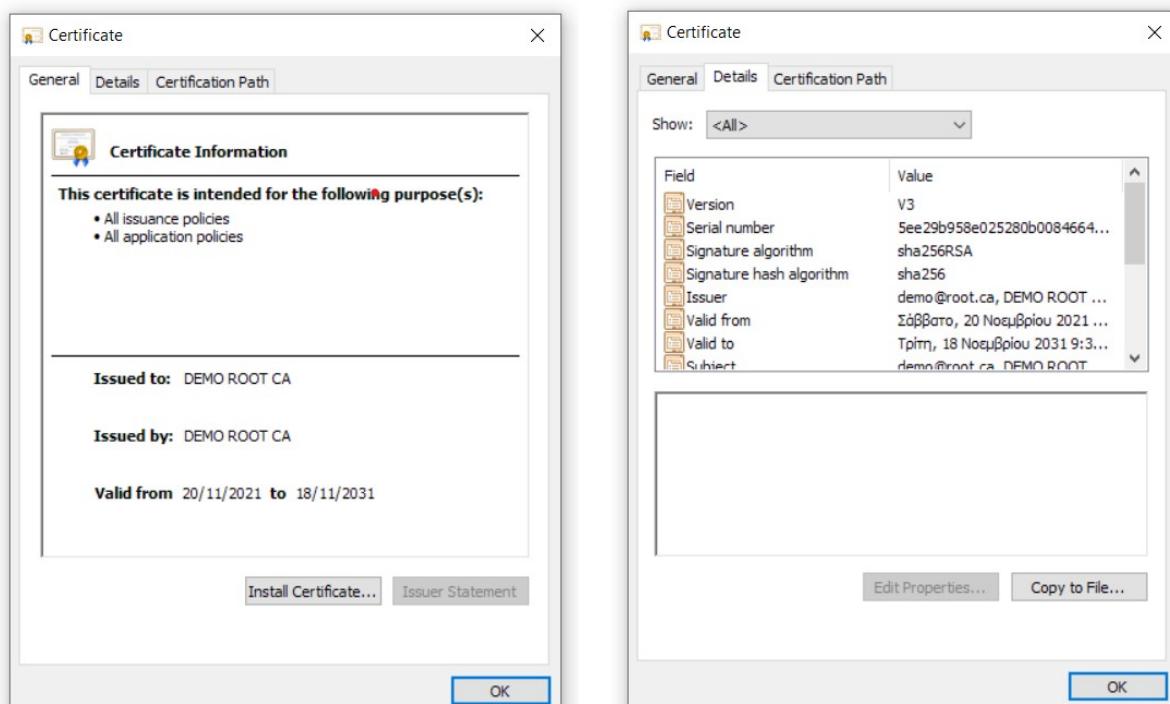
- Την **έκδοση** του **προτύπου** βάσει του οποίου έχει εκδοθεί.
- Τον **σειριακό αριθμό** – έναν μοναδικό αριθμό για κάθε πιστοποιητικό που εκδίδεται από έναν συγκεκριμένο ΠΥΕ (το όνομα του εκδότη και ο σειριακός αριθμός προσδιορίζουν ένα μοναδικό πιστοποιητικό).
- Τον **αλγόριθμο υπογραφής** – ένα αναγνωριστικό για τον αλγόριθμο που χρησιμοποιείται από τον ΠΥΕ για την υπογραφή του πιστοποιητικού.
- Τον **εκδότη** – προσδιορίζει την οντότητα που έχει υπογράψει και εκδώσει το πιστοποιητικό. Το πεδίο εκδότη πρέπει να περιέχει ένα διακεκριμένο όνομα (Distinguished Name – DN), ως όνομα τύπου X.501. Χαρακτηριστικά που αποτελούν μέρος του ονόματος του εκδότη είναι:
 - Country (C) – Ο κωδικός (2 γραμμάτων) της χώρας στην οποία δραστηριοποιείται ο εκδότης (π.χ. GR για την Ελλάδα).
 - Organisation (O) – Το όνομα του οργανισμού που εκδίδει το πιστοποιητικό.
 - Organisation Unit (OU) – Η οργανωτική μονάδα μέσα στον οργανισμό που εκδίδει το πιστοποιητικό.
 - Common Name (CN) – Ένα κοινό όνομα με το οποίο αναγνωρίζεται ο εκδότης.
- Την **περίοδο ισχύος** του πιστοποιητικού – το χρονικό διάστημα για το οποίο ο ΠΥΕ εγγυάται ότι θα διατηρήσει πληροφορίες σχετικά με την κατάσταση του πιστοποιητικού. Το πεδίο περιλαμβάνει την ακόλουθα δύο ημερομηνίων: την ημερομηνία κατά την οποία αρχίζει η περίοδος ισχύος του πιστοποιητικού (notBefore) και την ημερομηνία κατά την οποία λήγει η περίοδος ισχύος του πιστοποιητικού (notAfter).
- Τον **κάτοχο** – προσδιορίζει την οντότητα που σχετίζεται με το δημόσιο κλειδί που είναι αποθηκευμένο στο πεδίο δημόσιου κλειδιού κατόχου. Χαρακτηριστικά που αποτελούν μέρος του ονόματος του κατόχου είναι:
 - Country (C) – Ο κωδικός (2 γραμμάτων) της χώρας στην οποία δραστηριοποιείται ο εκδότης (π.χ. GR για την Ελλάδα).
 - Organisation (O) – Ο οργανισμός με τον οποίο σχετίζεται ο κάτοχος του πιστοποιητικού.

- Organisation Unit (OU) – Η οργανωτική μονάδα μέσα στον οργανισμό του κατόχου με την οποία σχετίζεται ο κάτοχος.
- Common Name (CN) – Ένα κοινό όνομα με το οποίο αναγνωρίζεται ο κάτοχος του πιστοποιητικού (π.χ. «Ιωάννης Παπαδόπουλος»).
- Το δημόσιο κλειδί του κατόχου του πιστοποιητικού.
- Τυχόν επεκτάσεις για συμπληρωματικές πληροφορίες που αφορούν το πιστοποιητικό, και
- Την ψηφιακή υπογραφή του ΠΥΕ στο πιστοποιητικό.

Ανάμεσα στις βασικές επεκτάσεις ενός πιστοποιητικού περιλαμβάνονται οι ακόλουθες:

- Πολιτικές πιστοποιητικών – οι όροι και οι προϋποθέσεις υπό τις οποίες λειτουργεί ο ΠΥΕ (π.χ. μέτρα ασφαλείας που σχετίζονται με πιστοποιητικά, κτλ.).
- Σημεία διανομής λίστας ανακληθέντων πιστοποιητικών (Certificate Revocation List Distribution Points) – προσδιορίζει το σημείο (URL) όπου μπορούν να βρεθούν πληροφορίες για τα ανακληθέντα πιστοποιητικά.

Στο Σχήμα 7.2 απεικονίζεται ένα παράδειγμα πιστοποιητικού (γενικές πληροφορίες αλλά και λεπτομέρειες) όπως αυτό εμφανίζεται στο λειτουργικό σύστημα Windows. Όπως παρατηρείτε, τα πεδία “Issued to” και “Issued by” περιέχουν ακριβώς τις ίδιες πληροφορίες. Αυτό υποδηλώνει αλλά και ταυτόχρονα συμβαίνει γιατί το πιστοποιητικό αυτό είναι ένα αυτο-υπογεγραμμένο πιστοποιητικό ΑΠ Ρίζας.



(α) Γενικές πληροφορίες.

(β) Λεπτομέρειες.

Σχήμα 7.2: Παράδειγμα ψηφιακού πιστοποιητικού Αρχής Πιστοποίησης Ρίζας.

7.1.3 Έλεγχος Κατάστασης Πιστοποιητικών

Τα πιστοποιητικά που ναι μεν δεν έχουν λήξει αλλά θα πρέπει για κάποιον λόγο να μη χρησιμοποιούνται άλλο, θα πρέπει να ανακαλούνται από τον εκάστοτε ΠΥΕ που τα έχει εκδώσει, και η πληροφορία αυτή θα πρέπει να διαχέται κατάλληλα προς όλους τους ενδιαφερόμενους και τα βασιζόμενα μέρη.

Τα πιστοποιητικά ανακαλούνται από τον ΠΥΕ, μεταξύ άλλων, για τους εξής λόγους:

- Ο κωδικός πρόσβασης στο ιδιωτικό κλειδί έχει γίνει γνωστός.
- Ο κάτοχος του πιστοποιητικού εγκαταλείπει τον οργανισμό για τον οποίο έχει εκδοθεί το πιστοποιητικό.
- Ο κάτοχος έχει απολέσει την κάρτα στην οποία είναι αποθηκευμένο το ιδιωτικό του κλειδί.
- Ο κάτοχος υποψιάζεται πως κάποιος τρίτος έχει αποκτήσει πρόσβαση στο ιδιωτικό του κλειδί.

Υπάρχουν δύο βασικοί μηχανισμοί που υποστηρίζουν τον έλεγχο κατάστασης πιστοποιητικών: οι Λίστες Ανακληθέντων Πιστοποιητικών και το Ηλεκτρονικό Πρωτόκολλο Κατάστασης Πιστοποιητικού.

7.1.3.1 Λίστες Ανακληθέντων Πιστοποιητικών

Στην περίπτωση της Λίστας Ανακληθέντων Πιστοποιητικών – ΛΑΠ (Certificate Revocation List) όλα τα ανακληθέντα πιστοποιητικά παρατίθενται σε μια λίστα η οποία εκδίδεται από τον ΠΥΕ που είναι υπεύθυνος για τη διαχείρισή τους.

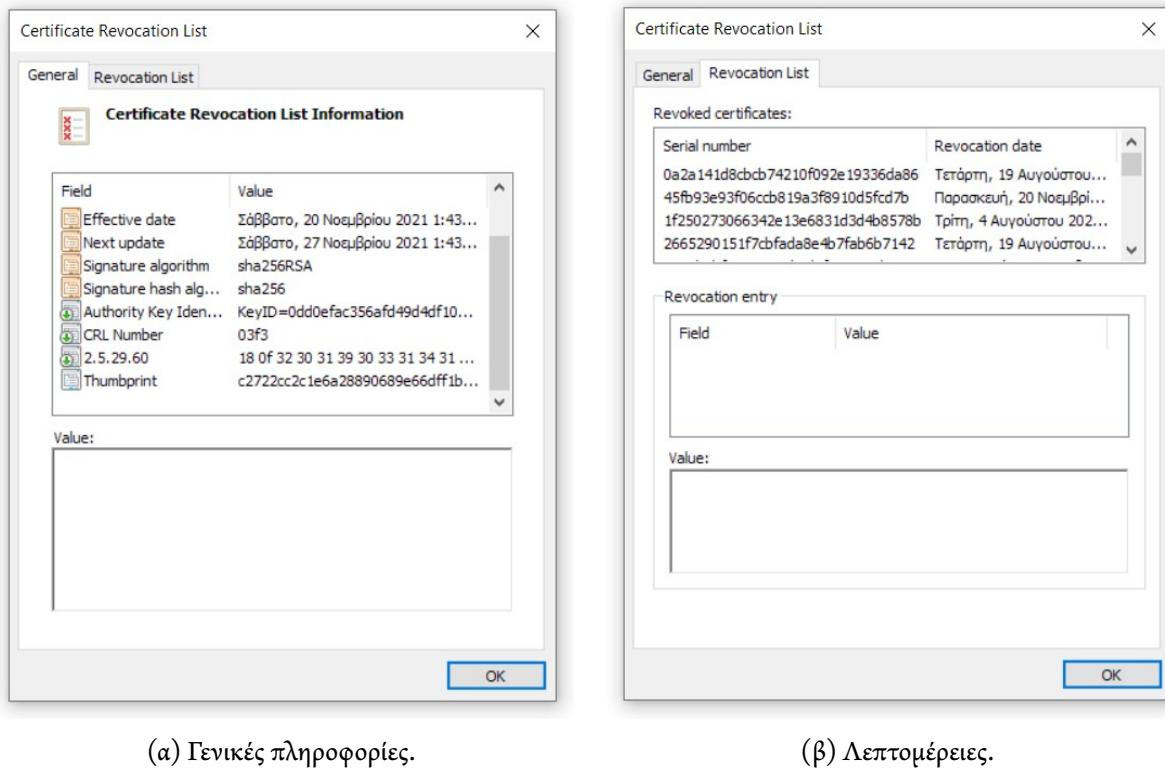
Σύμφωνα με την έκδοση 2.0 του ΛΑΠ προτύπου, όπως αυτό αποτυπώνεται στο RFC 5280 [1], μια λίστα περιέχει τα εξής:

- Την έκδοση του
- Τον αλγόριθμο που έχει χρησιμοποιηθεί για την υπογραφή της λίστας
- Το όνομα εκδότη – το όνομα του ΠΥΕ
- Την ημερομηνία έκδοσης της τρέχουσας λίστας
- Την ημερομηνία έκδοσης της επόμενης λίστας
- Τους αριθμούς σειράς των ανακληθέντων πιστοποιητικών
- Τυχόν επεκτάσεις
- Την ψηφιακή υπογραφή του ΠΥΕ για τη λίστα

Ένα ανακληθέν πιστοποιητικό παραμένει στη λίστα ανάκλησης τουλάχιστον μέχρι τη λήξη της περιόδου ισχύος του. Η πιο πρόσφατη λίστα ανάκλησης πρέπει να διατίθεται στους συμμετέχοντες στην ΥΔΚ σε περιοδική βάση, προκειμένου να διασφαλίζεται ότι μπορούν να επαληθεύουν την αξιοπιστία ενός πιστοποιητικού ανά πάσα στιγμή και να αποτρέπουν την κατάχρηση από κακόβουλους χρήστες που χρησιμοποιούν παράνομα το πιστοποιητικό κάποιου άλλου, καθώς και το σχετικό ιδιωτικό κλειδί.

Προβλήματα που παρουσιάζουν οι Λίστες Ανακληθέντων Πιστοποιητικών:

1. Όταν ανακαλείται ένα πιστοποιητικό, πρέπει να περιμένει κανείς μέχρι την επόμενη έκδοση της λίστας για να δημοσιοποιήσει αυτό το γεγονός. Αυτή η καθυστέρηση μπορεί να κυμαίνεται από μερικές ώρες έως αρκετές εβδομάδες.
2. Το μέγεθος της λίστας εξαρτάται από τον αριθμό των χρηστών μιας Αρχής Πιστοποίησης καθώς και από την περίοδο ισχύος των πιστοποιητικών.



(a) Γενικές πληροφορίες.

(b) Λεπτομέρειες.

Σχήμα 7.3: Παράδειγμα Λίστας Ανακληθέντων Πιστοποιητικών.

7.1.3.2 Ηλεκτρονικό Πρωτόκολλο Κατάστασης Πιστοποιητικού

Το Ηλεκτρονικό Πρωτόκολλο Κατάστασης Πιστοποιητικού – ΗΠΚΠ (Online Certificate Status Protocol – OCSP) χρησιμοποιείται για την ενημέρωση ενός βασιζόμενου μέρους σχετικά με την κατάσταση ανάκλησης ενός ψηφιακού πιστοποιητικού X.509. Περιγράφεται στο RFC 6960 [4]. Δημιουργήθηκε ως εναλλακτική λύση στις λίστες ανακληθέντων πιστοποιητικών, αντιμετωπίζοντας ορισμένα προβλήματα που σχετίζονται με τη χρήση ΛΑΠ σε μια υποδομή δημόσιου κλειδιού.

Όταν ένας χρήστης ζητά να μάθει την εγκυρότητα ενός πιστοποιητικού, αποστέλλεται ένα αίτημα (request) ΗΠΚΠ σε έναν αποκριτή (responder) ΗΠΚΠ. Αυτός ελέγχει την κατάσταση του συγκεκριμένου πιστοποιητικού και αποστέλλει πίσω μια απάντηση (response) ΗΠΚΠ με την οποία δηλώνει την κατάσταση του και η οποία μπορεί να πάρει τιμές: καλή (good), ανακληθέν (revoked) ή άγνωστη (unknown). Τα μηνύματα που κοινοποιούνται μέσω ΗΠΚΠ κωδικοποιούνται στο ASN.1 και συνήθως κοινοποιούνται μέσω ΗΤΤΡ.

7.1.4 Υπηρεσίες Ανάκτησης Κλειδιών

Σε πολλές περιπτώσεις, οι ΥΔΚ παρέχουν υπηρεσίες ανάκτησης κλειδιών για την υποστήριξη της επιχειρηματικής συνέχειας. Οι υπηρεσίες ανάκτησης κλειδιών αποθηκεύουν ιδιωτικά κλειδιά που χρησιμοποιούνται για την αποκρυπτογράφηση, για να διασφαλίσουν ότι το απλό κείμενο των κρυπτογραφημένων δεδομένων μπορεί να ανακτηθεί στο μέλλον. Αυτές οι υπηρεσίες μπορούν να παρέχουν το ιδιωτικό κλειδί στον χρήστη σε περίπτωση απώλειας ή αποτυχίας της κρυπτογραφικής μονάδας ή όταν υπάρχουν πολιτικές ή νομικές απαιτήσεις. Όταν υποστηρίζεται, αυτή η υπηρεσία αφαιρεί ένα φόρτο διαχείρισης κλειδιού από εφαρμογές που υποστηρίζουν τη χρήση κρυπτογραφίας δημοσίου κλειδιού. Η απαίτηση για ανάκτηση κλειδιών υπογραφής δεν είναι τόσο ισχυρή καθώς αυτά τα κλειδιά μπορούν να αντικατασταθούν με νέα δίχως επιπτώσεις στην επιχειρηματική συνέχεια.

7.1.5 Τύποι και Μεγέθη Κλειδιών

Ο Πίνακας 7.1 συνοψίζει τα προτεινόμενα κατά NIST [5] μεγέθη κλειδιών για ζεύγη κλειδιών που χρησιμοποιούνται από χρήστες μιας ΥΔΚ και για τις ανάγκες της υποδομής. Μια ΥΔΚ χρησιμοποιεί τον όρο κλειδί ψηφιακής υπογραφής (digital signature key) για να αναφέρεται σε ένα ιδιωτικό κλειδί υπογραφής ή ένα δημόσιο κλειδί επαλήθευσης υπογραφής που χρησιμοποιείται για τις ανάγκες παροχής μιας υπηρεσίας μη-αποποίησης. Ο όρος κλειδί αυθεντικοποίησης (authentication key) χρησιμοποιείται για να αναφέρεται σε ιδιωτικό ή δημόσιο κλειδί ελέγχου ταυτότητας. Σημειώστε ότι τόσο ένα κλειδί ψηφιακής υπογραφής όσο και ένα κλειδί αυθεντικοποίησης μπορούν να χρησιμοποιηθούν από έναν αλγόριθμο ψηφιακής υπογραφής.

Ένα κλειδί εγκαθίδρυσης κλειδιού (key establishment key) είναι ένα ζεύγος κλειδιών που χρησιμοποιείται για την παροχή συμφωνίας κλειδιού ή μεταφοράς κλειδιού. Ένα κλειδί υπογραφής της απόκρισης της ΑΠ και του αποκριτή ΗΠΚΠ είναι ένα κλειδί που χρησιμοποιείται για την υπογραφή και τον έλεγχο κατάστασης πιστοποιητικών.

Πίνακας 7.1: Τύποι και μεγέθη κλειδιών.

Τύπος Κλειδιού	Αλγόριθμοι και Μεγέθη Κλειδιών
Κλειδί ψηφιακής υπογραφής για έλεγχο ταυτότητας (αυθεντικοποίηση)	RSA (2048 bits) ECDSA (Curve P-256)
Κλειδί ψηφιακής υπογραφής για μη αποποίηση	RSA (2048 bits) ECDSA (Curve P-256 ή P-384)
Κλειδί Αρχής Πιστοποίησης και αποκριτή ΗΠΚΠ	RSA (2048 ή 3072 bits) ECDSA (Curve P-256 ή P-384)
Κλειδιά εγκαθίδρυσης κλειδιών (για χρήστες ή συσκευές)	RSA (2048 bits) Diffie-Hellman (2048 bits) ECDH (Curves P-256 ή P-384)

7.2 eIDAS

Γνωστός ως eIDAS (Electronic Identification and Trust Services) είναι ο Κανονισμός 910/2014 της ΕΕ για τις υπηρεσίες ηλεκτρονικής αναγνώρισης, ελέγχου ταυτότητας και εμπιστοσύνης [6]. Εγκρίθηκε από την Ευρωπαϊκή Επιτροπή το 2014 και είχε άμεσο αντίκτυπο στα κράτη μέλη της ΕΕ από το 2016. Ο κανονισμός eIDAS στοχεύει να ενισχύσει την εμπιστοσύνη στις ηλεκτρονικές συναλλαγές και να επιτρέψει τη διασυνοριακή αναγνώριση ηλεκτρονικών αναγνωριστικών, την ηλεκτρονική επαλήθευση ταυτότητας και τον έλεγχο ταυτότητας ηλεκτρονικών εγγράφων.

Βάσει του eIDAS είναι δυνατή η χρήση αυτών των υπηρεσιών εμπιστοσύνης καθώς και ηλεκτρονικών εγγράφων ως αποδεικτικά στοιχεία σε νομικές διαδικασίες σε όλα τα κράτη μέλη της ΕΕ που συμβάλλουν στη γενική διασυνοριακή χρήση τους. Τα δικαστήρια (ή άλλα όργανα που είναι αρμόδια για νομικές διαδικασίες) δεν μπορούν να τα απορρίψουν ως αποδεικτικά στοιχεία μόνο επειδή είναι ηλεκτρονικά, αλλά πρέπει να αξιολογήσουν αυτά τα ηλεκτρονικά εργαλεία με τον ίδιο τρόπο που θα έκαναν για το ισοδύναμό τους σε χαρτί. Ο κανονισμός eIDAS θεσπίζει την αρχή ότι ένα ηλεκτρονικό έγγραφο δεν θα πρέπει να απορρίπτεται με την αιτιολογία ότι είναι σε ηλεκτρονική μορφή.

Είτε πρόκειται για μεγάλη εταιρεία, είτε μικρομεσαία επιχείριση, είτε πολίτη που θέλει να ολοκληρώσει μια ηλεκτρονική συναλλαγή σε άλλη χώρα της ΕΕ, π.χ. υποβολή προσφορών σε μια πρόσκληση ή εγγραφή ως φοιτητής σε άλλο κράτος μέλος της ΕΕ, εκτός από τη μείωση του χρόνου και του κόστους, ο κανονισμός eIDAS

εξασφαλίζει τη διασυνοριακή αναγνώριση των εθνικών ηλεκτρονικών ταυτότητων (eID) και των υπηρεσιών ηλεκτρονικής εμπιστοσύνης που υποστηρίζουν την ηλεκτρονική τους συναλλαγή. Ως εκ τούτου, ενισχύει την εμπιστοσύνη, την ασφάλεια και την ευκολία στη χρήση (διασυνοριακών) ηλεκτρονικών υπηρεσιών. Από την 1η Ιουλίου 2016, οι περισσότερες διατάξεις του κανονισμού eIDAS εφαρμόζονται άμεσα στο νομικό πλαίσιο των 28 κρατών μελών της ΕΕ για την αντιμετώπιση προβλημάτων κατακερματισμού των εθνικών νομοθετημάτων. Ο κανονισμός eIDAS διασφαλίζει ότι οι φυσικές και νομικές οντότητες μπορούν να χρησιμοποιούν τα εθνικές τους ηλεκτρονικές ταυτότητες για πρόσβαση σε δημόσιες υπηρεσίες σε άλλες χώρες της ΕΕ όπου απαιτούνται eIDs, καθιστώντας τες παράλληλα ισοδύναμες με τις παραδοσιακές διαδικασίες που βασίζονται σε χαρτί.

Ο κανονισμός eIDAS ορίζει ηλεκτρονικές υπηρεσίες εμπιστοσύνης που μπορούν να παρέχονται έναντι αμοιβής. Οι υπηρεσίες αυτές αφορούν:

- Την δημιουργία, εξακρίβωση και επικύρωση ηλεκτρονικών υπογραφών², ηλεκτρονικών σφραγίδων ή ηλεκτρονικών χρονοσφραγίδων, ηλεκτρονικών υπηρεσιών συστημάτων παράδοσης και πιστοποιητικών που σχετίζονται με τις υπηρεσίες αυτές.
- Την δημιουργία, εξακρίβωση και επικύρωση πιστοποιητικών για επαλήθευση της ταυτότητας ιστοτόπων.
- Την διαφύλαξη ηλεκτρονικών υπογραφών, σφραγίδων ή πιστοποιητικών που σχετίζονται με τις υπηρεσίες αυτές.

Προκειμένου να διασφαλιστεί υψηλού επιπέδου ασφάλεια των πιστοποιημένων υπηρεσιών εμπιστοσύνης, ο κανονισμός eIDAS προβλέπει ένα σύστημα ενεργού εποπτείας των Εγκεκριμένων Παρόχων Υπηρεσιών Εμπιστοσύνης – ΕΠΥΕ (Qualified Trust Service Provider) και των Εγκεκριμένων Υπηρεσιών Εμπιστοσύνης – ΕΥΕ (Qualified Trust Service) που παρέχουν (εφεξής καλούμενες ως ΕΠΥΕ/ΕΥΕ) από τον εθνικό αρμόδιο εποπτικό φορέα που εποπτεύει, εκ των προτέρων και εκ των υστέρων, την εκπλήρωση των απαιτήσεων και υποχρεώσεων των ΕΠΥΕ/ΕΥΕ.

Ο Πίνακας 7.2 απεικονίζει τις ιδιότητες ασφάλειας που έχουν οι πέντε εγκεκριμένες υπηρεσίες του eIDAS, δηλαδή οι εγκεκριμένες ηλεκτρονικές υπογραφές (Qualified Electronic Signatures – QES), οι εγκεκριμένες ηλεκτρονικές σφραγίδες (Qualified Electronic Seals – QESeal), οι εγκεκριμένες ηλεκτρονικές χρονοσφραγίδες (Qualified Electronic Time Stamps – QETS), οι εγκεκριμένες ηλεκτρονικές υπηρεσίες συστημάτων παράδοσης (Qualified Electronic Registered Delivery – QeDel), και τα εγκεκριμένα πιστοποιητικά για επαλήθευση της ταυτότητας ιστοτόπων (Qualified Certificates for Website Authentication – QWAC).

Εξυπηρετώντας καθαρά σκοπούς μάρκετινγκ, αφού πιστοποιηθεί, ένας ΕΠΥΕ/ΕΥΕ μπορεί να χρησιμοποιήσει το σήμα εμπιστοσύνης της ΕΕ για εξειδικευμένες υπηρεσίες εμπιστοσύνης με σκοπό την προώθηση των εγκεκριμένων υπηρεσιών εμπιστοσύνης που παρέχει.

Η χρήση του σήματος εμπιστοσύνης της ΕΕ, η οποία είναι εθελοντική, αποσκοπεί στην ενίσχυση της διαφάνειας της αγοράς και στη βοήθεια των καταναλωτών να κάνουν διάκριση μεταξύ εγκεκριμένων υπηρεσιών εμπιστοσύνης και μη.

² Είναι χρήσιμο να διευκρινήσουμε τη διαφορά μεταξύ «ηλεκτρονικής» και «ψηφιακής» υπογραφής. Μία «ηλεκτρονική υπογραφή» είναι μια νομική έννοια που ορίζεται στο eIDAS [6] ως εξής: «δεδομένα σε ηλεκτρονική μορφή τα οποία είναι συνημμένα σε άλλα ηλεκτρονικά δεδομένα ή συσχετίζονται λογικά με άλλα δεδομένα σε ηλεκτρονική μορφή και τα οποία χρησιμοποιούνται από τον υπογράφοντα για να υπογράφει». Από την άλλη πλευρά, η «ψηφιακή υπογραφή» αναφέρεται σε μία μαθηματική και κρυπτογραφική έννοια που χρησιμοποιείται ευρέως για να παρέχει συγκεκριμένα και πρακτικά παραδείγματα ηλεκτρονικής υπογραφής. Ο ορισμός που δίνεται από το ETSI TR 119 100 [7] είναι «δεδομένα που επισυνάπτονται ή είναι το αποτέλεσμα κρυπτογραφικού μετασχηματισμού μιας μονάδας δεδομένων, επιτρέποντας στον παραλήπτη της μονάδας δεδομένων να αποδείξει την πηγή και την ακεραιότητα της μονάδας δεδομένων και να προστατευτεί από πλαστογραφία, π.χ. από τον παραλήπτη». Αυτές οι δύο έννοιες πρέπει να διαχωρίζονται, καθώς όλες οι ηλεκτρονικές υπογραφές δεν είναι απαραίτητα ψηφιακές υπογραφές.

Πίνακας 7.2: Συγκριτικός πίνακας λειτουργιών που προσφέρονται από τους διάφορους τύπους εγκεκριμένων υπηρεσιών εμπιστοσύνης.

Ακεραιότητα Δεδομένων	Εμπιστευτικότητα	Αυθεντικοποίηση Πηγής (Φυσικά Πρόσωπα)	Αυθεντικοποίηση Πηγής (Νομικά Πρόσωπα)	Πιστοποίηση Χρόνου
QES	✓	✗	✓	✗
QESeal	✓	✗	✗	✓
QETS	✓	✗	✗	✓
QeDel	✓	✓	✓	✓
QWAC	✓	✓	✓	✗



Σχήμα 7.4: Ευρωπαϊκό σήμα εμπιστοσύνης για τις εγκεκριμένες υπηρεσίες εμπιστοσύνης.

7.2.1 Τύποι Ηλεκτρονικών Υπογραφών κατά eIDAS

Ο κανονισμός eIDAS ορίζει τρεις τύπους ηλεκτρονικών υπογραφών: ηλεκτρονικές υπογραφές, προηγμένες ηλεκτρονικές υπογραφές, και εγκεκριμένες ηλεκτρονικές υπογραφές.

7.2.1.1 Ηλεκτρονικές Υπογραφές

Στην πιο βασική της μορφή, η ηλεκτρονική υπογραφή ορίζεται ως «δεδομένα σε ηλεκτρονική μορφή τα οποία είναι συνημμένα σε άλλα ηλεκτρονικά δεδομένα ή συσχετίζονται λογικά με άλλα δεδομένα σε ηλεκτρονική μορφή και τα οποία χρησιμοποιούνται από τον υπογράφοντα για να υπογράφει». Με αυτόν τον ορισμό, αν και πολύ γενικός ώστε να καλύπτει ένα ευρύ φάσμα λύσεων, ο κανονισμός eIDAS θέτει τα θεμέλια για όλες τις ηλεκτρονικές υπογραφές, ενώ θεσπίζει την αρχή ότι «δεν θα πρέπει να απορρίπτεται η ισχύς της ηλεκτρονικής υπογραφής με την αιτιολογία ότι είναι σε ηλεκτρονική μορφή ή ότι δεν πληροί τις απαιτήσεις της εγκεκριμένης ηλεκτρονικής υπογραφής». Ωστόσο, αυτού του τύπου οι ηλεκτρονικές υπογραφές δεν μπορούν να αποδείξουν την πραγματική ταυτότητα του ατόμου που την υπέγραψε και δεν μπορούν να εγγυηθούν ότι το υπογεγραμμένο έγγραφο δεν έχει τροποποιηθεί.

7.2.1.2 Προηγμένες Ηλεκτρονικές Υπογραφές

Η προηγμένη ηλεκτρονική υπογραφή είναι ηλεκτρονική υπογραφή η οποία πληροί τις ακόλουθες απαιτήσεις:

- Συνδέεται κατά τρόπο μοναδικό με τον υπογράφοντα.
- Είναι ικανή να ταυτοποιεί τον υπογράφοντα.

- Δημιουργείται με δεδομένα δημιουργίας ηλεκτρονικής υπογραφής τα οποία ο υπογράφων μπορεί, με υψηλό βαθμό εμπιστοσύνης, να χρησιμοποιεί υπό τον αποκλειστικό του έλεγχο.
- Συνδέεται με τα δεδομένα που έχουν υπογραφεί σε σχέση με αυτήν κατά τρόπο ώστε να μπορεί να ανιχνευθεί οποιαδήποτε επακόλουθη τροποποίηση των εν λόγω δεδομένων.

Οι προηγμένες ηλεκτρονικές υπογραφές αυξάνουν το επίπεδο ασφάλειας, καθώς η υπογραφή συνδέεται πιο αξιόπιστα με το άτομο που υπογράφει το ηλεκτρονικό έγγραφο. Ωστόσο, δεν παρέχουν το βέλτιστο επίπεδο αξιοπιστίας που παρέχεται από τον ακόλουθο τύπο υπογραφών, τις εγκεκριμένες ηλεκτρονικές υπογραφές.

7.2.1.3 Εγκεκριμένες Ηλεκτρονικές Υπογραφές

Σήμερα, είναι δυνατή η ηλεκτρονική υπογραφή δεδομένων και η επίτευξη των ίδιων αποτελεσμάτων με αυτά της χρήσης ιδιόχειρης υπογραφής. Τέτοιες ηλεκτρονικές υπογραφές που επωφελούνται από πλήρη νομική αναγνώριση χάρη στον κανονισμό eIDAS ονομάζονται εγκεκριμένες ηλεκτρονικές υπογραφές. Μια εγκεκριμένη ηλεκτρονική υπογραφή ορίζεται ως «η προηγμένη ηλεκτρονική υπογραφή που δημιουργείται από εγκεκριμένη διάταξη δημιουργίας ηλεκτρονικής υπογραφής και η οποία βασίζεται σε εγκεκριμένο πιστοποιητικό ηλεκτρονικής υπογραφής».

Οι εγκεκριμένες ηλεκτρονικές υπογραφές απαιτούν τη χρήση πιο εξελιγμένης τεχνολογίας για την αποθήκευση των ιδιωτικών κλειδιών και τη δημιουργία ηλεκτρονικών υπογραφών, όπως έξυπνες κάρτες, φορητές συσκευές ή μονάδες ασφαλείας υλισμικού που πληρούν συγκεκριμένα πρότυπα ασφαλείας. Επίσης υποστηρίζονται από ένα πιο λεπτομερές πιστοποιητικό γνωστό ως εγκεκριμένο πιστοποιητικό ηλεκτρονικής υπογραφής (qualified electronic signature certificate), που εκδίδεται από έναν εγκεκριμένο πάροχο υπηρεσιών εμπιστοσύνης. Λόγω αυτών των απαιτήσεων, οι εγκεκριμένες ηλεκτρονικές υπογραφές αναγνωρίζονται ρητά ότι έχουν νομική ισχύ ισοδύναμη με τις ιδιόχειρες υπογραφές.

Τα εγκεκριμένα πιστοποιητικά ηλεκτρονικής υπογραφής, τα οποία εκδίδονται μόνο σε φυσικά πρόσωπα, περιέχουν πληροφορίες, μεταξύ άλλων, σχετικά με τον κάτοχο του πιστοποιητικού, την περίοδο ισχύος του, την τοποθεσία των υπηρεσιών που παρέχουν πληροφορίες κατάστασης και μια ένδειξη πως το πιστοποιητικό είναι εγκεκριμένο.

Το εγκεκριμένο πιστοποιητικό ηλεκτρονικής υπογραφής πληροί τις ακόλουθες απαιτήσεις:

- Ένδειξη, τουλάχιστον σε μορφή κατάλληλη για αυτοματοποιημένη επεξεργασία, ότι το πιστοποιητικό έχει εκδοθεί ως εγκεκριμένο πιστοποιητικό ηλεκτρονικής υπογραφής.
- Ένα σύνολο δεδομένων που αντιπροσωπεύουν αναμφίσημα τον εγκεκριμένο πάροχο υπηρεσιών εμπιστοσύνης ο οποίος έχει εκδώσει τα εγκεκριμένα πιστοποιητικά και περιλαμβάνουν τουλάχιστον το κράτος μέλος στο οποίο είναι εγκατεστημένος και σε περίπτωση που πρόκειται για νομικό πρόσωπο: το όνομα και, κατά περίπτωση, τον αριθμό μητρώου του, όπως αναφέρεται στα επίσημα αρχεία, ενώ σε περίπτωση που πρόκειται για φυσικό πρόσωπο: το όνομα του προσώπου.
- Τουλάχιστον το όνομα του υπογράφοντος ή ένα ψευδώνυμο – εάν χρησιμοποιείται ψευδώνυμο, πρέπει να αναφέρεται σαφώς.
- Δεδομένα επικύρωσης ηλεκτρονικής υπογραφής που αντιστοιχούν στα δεδομένα δημιουργίας ηλεκτρονικής υπογραφής.
- Λεπτομέρειες για την έναρξη και τη λήξη της περιόδου ισχύος του πιστοποιητικού.
- Τον κωδικό ταυτότητας του πιστοποιητικού, ο οποίος πρέπει να είναι μοναδικός για τον εγκεκριμένο πάροχο υπηρεσιών εμπιστοσύνης.
- Την προηγμένη ηλεκτρονική υπογραφή ή την προηγμένη ηλεκτρονική σφραγίδα του εκδίδοντος εγκεκριμένου παρόχου υπηρεσιών εμπιστοσύνης.

- Την τοποθεσία όπου διατίθεται δωρεάν το πιστοποιητικό το οποίο τεκμηριώνει την προηγμένη ηλεκτρονική υπογραφή ή την προηγμένη ηλεκτρονική σφραγίδα που αναφέρεται στο προηγούμενο στοιχείο.
- Την τοποθεσία των υπηρεσιών που μπορούν να χρησιμοποιηθούν για την άντληση πληροφοριών σχετικά με το καθεστώς ισχύος του εγκεκριμένου πιστοποιητικού.
- Σε περίπτωση που τα δεδομένα δημιουργίας ηλεκτρονικής υπογραφής τα οποία σχετίζονται με τα δεδομένα επικύρωσης ηλεκτρονικής υπογραφής βρίσκονται σε εγκεκριμένη διάταξη δημιουργίας ηλεκτρονικής υπογραφής, κατάλληλη σχετική ένδειξη, τουλάχιστον σε μορφή κατάλληλη για αυτοματοποιημένη επεξεργασία.

7.2.1.4 Εγκεκριμένες Διατάξεις Δημιουργίας Ηλεκτρονικής Υπογραφής

Οι εγκεκριμένες διατάξεις δημιουργίας ηλεκτρονικής υπογραφής (qualified electronic signature creation device) που απαιτεί ο κανονισμός eIDAS για τη δημιουργία εγκεκριμένων ηλεκτρονικών υπογραφών, είναι συσκευές που διασφαλίζουν, με κατάλληλα τεχνικά και διαδικαστικά μέσα, ότι τουλάχιστον το ιδιωτικό κλειδί παραμένει προστατευμένο και υπό τον πλήρη έλεγχο του νόμιμου υπογράφοντα. Σύμφωνα με τον κανονισμό eIDAS, οι εγκεκριμένες διατάξεις δημιουργίας ηλεκτρονικής υπογραφής θα πρέπει να ικανοποιούν τις ακόλουθες απαιτήσεις:

- Να διασφαλίζουν ευλόγως την εμπιστευτικότητα των δεδομένων δημιουργίας ηλεκτρονικής υπογραφής.
- Τα δεδομένα δημιουργίας ηλεκτρονικής υπογραφής που χρησιμοποιούνται για τη δημιουργία της ηλεκτρονικής υπογραφής να προκύπτουν στην πράξη μία μόνο φορά.
- Τα δεδομένα δημιουργίας ηλεκτρονικής υπογραφής να μην μπορούν, με εύλογη βεβαιότητα, να παραχθούν και ότι η ηλεκτρονική υπογραφή προστατεύεται με τρόπο αξιόπιστο από πλαστογραφία με τη χρήση της τρέχουσας τεχνολογίας.
- Τα δεδομένα δημιουργίας ηλεκτρονικής υπογραφής μπορούν να προστατεύονται κατά τρόπο αξιόπιστο από τον νόμιμο υπογράφοντα έναντι της χρησιμοποίησής τους από τρίτους.

Τυπικές συσκευές που πληρούν αυτές τις απαιτήσεις είναι οι έξυπνες κάρτες και οι μονάδες ασφαλείας υλισμικού.

- Οι έξυπνες κάρτες (smart card) είναι συσκευές ανθεκτικές σε παραβιάσεις που έχουν τη δυνατότητα αποθήκευσης και επεξεργασίας δεδομένων με ασφάλεια. Περιλαμβάνουν ένα ενσωματωμένο τσιπ ολοκληρωμένου κυκλώματος (ICC) που μπορεί να είναι ένας ασφαλής μικροελεγκτής με εσωτερική μνήμη ή ένα τσιπ μνήμης μόνο. Η κάρτα συνδέεται με συσκευή ανάγνωσης με άμεση φυσική επαφή ή ανέπαφα, με απομακρυσμένη διεπαφή ραδιοσυχνοτήτων. Με έναν ενσωματωμένο μικροελεγκτή, οι έξυπνες κάρτες έχουν τη δυνατότητα να αποθηκεύουν ικανοποιητικές ποσότητες δεδομένων, να εκτελούν τις δικές τους λειτουργίες στην κάρτα (π.χ. κρυπτογράφηση και αμοιβαίο έλεγχο ταυτότητας) και να αλληλεπιδρούν έξυπνα με μια συσκευή ανάγνωσης έξυπνων καρτών. Η τεχνολογία έξυπνων καρτών συμμορφώνεται με τα διεθνή πρότυπα (ISO/IEC 7816 και ISO/IEC 14443) και είναι διαθέσιμη σε ποικίλες μορφές, όπως πλαστικές κάρτες, μονάδες ταυτότητας συνδρομητή (USIM) που χρησιμοποιούνται σε κινητά τηλέφωνα, και διάφορες USB συσκευές.
- Οι μονάδες ασφαλείας υλισμικού (Hardware Security Module – HSM), όπως έχουν αναλυθεί και στην Ενότητα 6.7, είναι συσκευές ασφαλείας που μπορούν να υποστηρίζουν τη δημιουργία υπογραφών από απόσταση καθώς τα ιδιωτικά κλειδιά αποθηκεύονται στο HSM και είναι προσβάσιμα μόνο

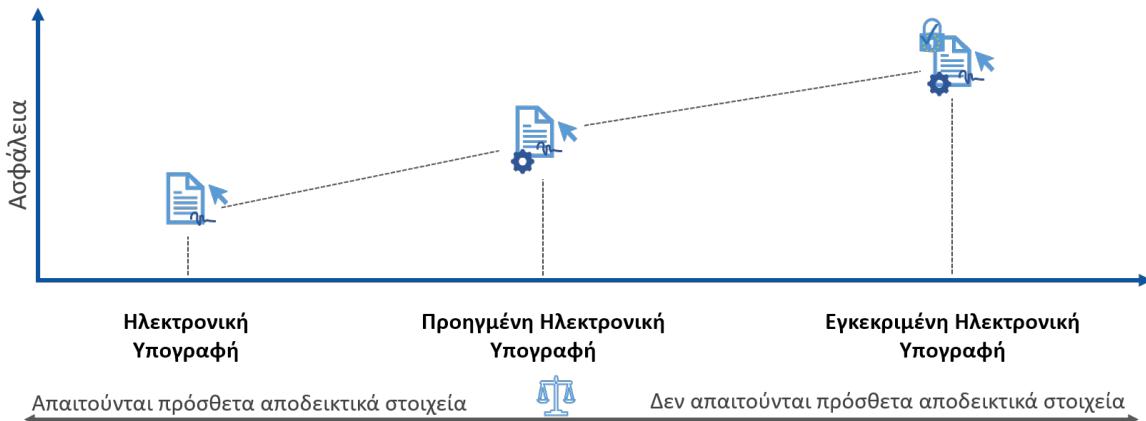
όταν ο κάτοχος του κλειδιού θέλει να δημιουργήσει μια υπογραφή. Με αυτόν τον τρόπο, είναι δυνατή η απόκτηση μέσω ενός εγκεκριμένου παρόχου, ενός εγκεκριμένου πιστοποιητικού στο cloud και η απομακρυσμένη χρήση του για τη δημιουργία εγκεκριμένων ηλεκτρονικών υπογραφών. Δεδομένου ότι το ηλεκτρονικό πιστοποιητικό είναι στο cloud, σωστά αποθηκευμένο και προστατευμένο, παρέχει υψηλό βαθμό κινητικότητας, καθώς η ηλεκτρονική υπογραφή μπορεί να δημιουργηθεί από το HSM, δηλαδή από οπουδήποτε.

Όλες αυτές οι συσκευές, στις οποίες γίνεται αποθήκευση των ιδιωτικών κλειδιών δημιουργίας υπογραφής πρέπει να πιστοποιηθούν βάσει ενός κοινά αποδεκτού σχήματος αξιολόγησης, όπως τα Common Criteria, για να θεωρηθούν κατάλληλες για χρήση.

7.2.2 Νομικός Ορισμός των Εγκεκριμένων Ηλεκτρονικών Υπογραφών

Λαμβάνοντας υπόψη τον ορισμό της, μια ηλεκτρονική υπογραφή μπορεί να δημιουργηθεί χρησιμοποιώντας πολλές τεχνολογίες, που κυμαίνονται από τις εξαιρετικά αδύναμες (όπως η επικόλληση μιας σαρωμένης εικόνας υπογραφής σε ένα έγγραφο) έως τις πολύ ισχυρές (όπως η υπογραφή με χρήση ηλεκτρονικής ταυτότητας με δακτυλικό αποτύπωμα).

Οι ηλεκτρονικές υπογραφές γενικά δεν στερούνται νομικού αποτελέσματος, ενώ χρησιμοποιούνται ως αποδεικτικό στοιχείο σε δικαστικές διαδικασίες. Εντός της οικογένειας ηλεκτρονικών υπογραφών, ο κανονισμός eIDAS ορίζει υποσύνολα ηλεκτρονικών υπογραφών που παρέχουν αυξανόμενη νομική προβλεψιμότητα μέχρι την εγκεκριμένη ηλεκτρονική υπογραφή, η οποία επωφελείται από τη νομική ισοδυναμία με τις χειρόγραφες υπογραφές, όπως απεικονίζεται στο Σχήμα 7.5.



Σχήμα 7.5: Οι εγκεκριμένες ηλεκτρονικές υπογραφές έχουν το ισοδύναμο νομικό αποτέλεσμα με τις χειρόγραφες υπογραφές.

Η νομική ισοδυναμία βασίζεται στο γεγονός ότι οι εγκεκριμένες ηλεκτρονικές υπογραφές υποστηρίζονται από (i) αξιόπιστες τεχνολογίες, παρόμοιες με αυτές των προηγμένων ηλεκτρονικών υπογραφών, (ii) αναγνωρισμένες συσκευές δημιουργίας ηλεκτρονικών υπογραφών, και (iii) εγκεκριμένους παρόχους υπηρεσιών εμπιστοσύνης (ΕΠΥΕ) που εποπτεύονται από τα κράτη μέλη της ΕΕ μέσω των αρμόδιων εποπτικών οργάνων.

7.2.2.1 Διεύρυνση Υπογραφών

Η διεύρυνση υπογραφών (long-term signature validation) αφορά τη διαδικασία με την οποία ένα συγκεκριμένο υποστηρικτικό υλικό, όπως χρονοσφραγίδες και δεδομένα επικύρωσης, ενσωματώνεται στις υπογραφές για να τις καταστήσει πιο ανθεκτικές στην αλλαγή ή για να διευρύνει τη μακροζωία τους. Πράγματι, όταν η

υπογραφή πρέπει να επικυρωθεί μετά τη δημιουργία της, είναι απαραίτητο να ελεγχθεί, π.χ. ότι το πιστοποιητικό δεν ανακλήθηκε κατά τη στιγμή της υπογραφής. Εάν υπήρξε ανάκληση μεταξύ της χρονικής στιγμής της δημιουργίας της υπογραφής και της στιγμής επικύρωσής της, το βασιζόμενο μέρος πρέπει να έχει επαρκή στοιχεία ότι η υπογραφή δημιουργήθηκε πριν τη στιγμή της ανάκλησης. Αυτή η διεύρυνση της μακροζωίας μπορεί να γίνει είτε από τον υπογράφοντα, είτε από το βασιζόμενο μέρος, είτε από πάροχο υπηρεσιών εμπιστοσύνης που επικυρώνει ή διατηρεί την υπογραφή εξ ονόματος του υπογράφοντος ή του βασιζόμενου μέρους.

Συνήθως, η ημερομηνία υπογραφής αναφέρεται στο υπογεγραμμένο έγγραφο, καθώς αυτό απαιτείται από την κείμενη νομοθεσία, αλλά αυτό μπορεί να μην είναι αρκετό. Για ηλεκτρονικά έγγραφα, η χρονοσήμανση είναι μια δυνατότητα αποφυγής κινδύνων παραποίησης. Η ηλεκτρονική χρονοσήμανση είναι δεδομένα σε ηλεκτρονική μορφή που δεσμεύουν άλλα ηλεκτρονικά δεδομένα σε μια συγκεκριμένη χρονική στιγμή αποδεικνύοντας ότι αυτά τα δεδομένα υπήρχαν εκείνη τη στιγμή. Η χρονοσήμανση της υπογραφής παρέχει την απόδειξη ότι δημιουργήθηκε πριν από την ημερομηνία που αναγράφεται στην ηλεκτρονική χρονοσήμανση. Αυτό επιτρέπει στο βασιζόμενο μέρος να αξιολογήσει την ημερομηνία δημιουργίας σε σχέση με την ημερομηνία πιθανής ανάκλησης.

Σε μια υπογραφή με δεδομένα μακροπρόθεσμης επικύρωσης (long term validation data), αποδεικτικά στοιχεία προστίθενται στην υπογραφή τα οποία βοηθούν στην επέκταση της χρονικής διάρκειας εγκυρότητας της υπογραφής. Το σύνολο του υλικού επικύρωσης ή των αναφορών σε αυτό αρκεί για να εξακριβωθεί η κατάσταση επικύρωσης όλων των πιστοποιητικών (όπως, πιστοποιητικά υπογράφοντος και χρονοσήμανσης) που συνοδεύουν το υπογεγραμμένο αρχείο. Η διαδικασία της επικύρωσης της χρονικής στιγμής δημιουργίας υπογραφής με τη χρήση χρονοσφραγίδας θα πρέπει να επαναλαμβάνεται εγκαίρως προτού η προστασία που παρέχεται από μια προηγούμενη χρονοσφραγίδα καταστεί μη αποτελεσματική, και θα πρέπει να χρησιμοποιεί ισχυρότερους αλγόριθμους ή μεγαλύτερα μήκη κλειδιών από αυτά που έχουν χρησιμοποιηθεί στις αρχικές υπογραφές ή τα διακριτικά της χρονοσφραγίδας.

Οι εγκεκριμένες ηλεκτρονικές υπογραφές μπορούν να υποστηριχθούν από πρόσθετες εγκεκριμένες υπηρεσίες εμπιστοσύνης, όπως είναι

- Εγκεκριμένες ηλεκτρονικές χρονοσφραγίδες (όπως ορίζονται στο άρθρο 42 του κανονισμού eIDAS).
- Εγκεκριμένη επικύρωση της εγκεκριμένης ηλεκτρονικής υπογραφής (όπως ορίζεται στο άρθρο 33 του κανονισμού eIDAS).
- Εγκεκριμένη διαφύλαξη της εγκεκριμένης ηλεκτρονικής υπογραφής (όπως ορίζεται στο άρθρο 34 του κανονισμού eIDAS).

7.2.2.2 Πρότυπα Σχετικά με τις Εγκεκριμένες Ηλεκτρονικές Υπογραφές

Μορφές Υπογραφών: Όταν ένας οργανισμός αποφασίζει να αναπτύξει τη δική του εφαρμογή δημιουργίας υπογραφών (είτε θα την αναπτύξει στο περιβάλλον των χρηστών του είτε θα την προσφέρει ως κεντρική υπηρεσία), η πρώτη σύσταση είναι να χρησιμοποιεί τυπικές και εγκεκριμένες μορφές υπογραφής, συγκεκριμένα αυτές αναφέρονται από το CID (ΕΕ) 2015/1506 σύμφωνα με το άρθρο 27 του κανονισμού eIDAS.

Τέτοιες εγκεκριμένες μορφές υπογραφής ορίζονται στα ακόλουθα πρότυπα:

- ETSI TS 103 171 (XAdES Baseline Profile) [8]: Το XAdES (XML Advanced Electronic Signatures), είναι μια επέκταση της προδιαγραφής XML-Signature Syntax and Processing (XMLDSig). Επικεντρώνεται στην ενσωμάτωση πρόσθετων χαρακτηριστικών κρυπτογράφησης και ασφάλειας που είναι απαραίτητα για τη δημιουργία νομικά αναγνωρισμένων ηλεκτρονικών υπογραφών. Το προφίλ βασικής γραμμής XAdES (ETSI TS 103 171) ορίζει ένα σύνολο κανόνων και κατευθυντήριων γραμμών που συμβάλλουν στη διασφάλιση της διαλειτουργικότητας και της ασφάλειας των προηγμένων ηλεκτρονικών υπογραφών σε διάφορες εφαρμογές και κλάδους.
- ETSI TS 103 172 (PAdES Baseline Profile) [9]: Το PAdES (PDF Advanced Electronic Signatures), στοχεύει να παρέχει μια τυποποιημένη προσέγγιση για την ενσωμάτωση προηγμένων ηλεκτρονικών

υπογραφών σε αρχεία PDF. Αυτό επιτρέπει την υπογραφή εγγράφων PDF με τρόπο που διασφαλίζει την ακεραιότητα, τη γνησιότητα και τη νομική τους εγκυρότητα.

- ETSI TS 103 173 (CAdES Baseline Profile) [10]: Το CAdES (CMS Advanced Electronic Signatures), είναι μια επέκταση του Cryptographic Message Syntax (CMS), το πρότυπο του IETF για κρυπτογραφικά προστατευμένα μηνύματα, το οποίο ορίζει ένα σύνολο μορφών και μηχανισμών για τη δημιουργία προηγμένων ηλεκτρονικών υπογραφών, οι οποίες είναι επαληθεύσιμες και συμβατές με νομικές και κανονιστικές απαιτήσεις.
- ETSI TS 103 174 (Associated Signature Container Baseline Profile) [11]: Το Associated Signature Container (ASiC) ορίζει μια τυποποιημένη προσέγγιση για τη δημιουργία και τη διαχείριση ηλεκτρονικών υπογραφών και σχετικών δεδομένων εντός μορφών container (μορφές πακέτων βασισμένες στο ZIP). Τα ASiC έχουν σχεδιαστεί για να διασφαλίζουν την ακεραιότητα, τη γνησιότητα και τη μακροπρόθεσμη ισχύ των ηλεκτρονικών υπογραφών και του περιεχομένου με το οποίο σχετίζονται.

Αυτά τα πρότυπα υποστηρίζουν διαφορετικές μορφές υπογραφών, οι οποίες είναι κατάλληλες για διαφορετικούς όρους διατήρησης (έως πολύ μακροπρόθεσμα). Συνιστάται στους φορείς υλοποίησης, όποτε είναι δυνατόν, να χρησιμοποιούν τις πιο προηγμένες φόρμες που επιτρέπουν τις καλύτερες εγγυήσεις όχι μόνο μακροπρόθεσμα, αλλά και σε περίπτωση πολλών τύπων παραβιάσεων ασφάλειας που ενδέχεται να συμβούν και μεσοπρόθεσμα.

Πολιτικές και Απαιτήσεις Ασφαλείας για Εφαρμογές που Παρέχουν Δημιουργία και Επικύρωση Υπογραφής: Το πρότυπο ETSI TS 119 101 [12] παρέχει γενικές απαιτήσεις ασφάλειας και πολιτικής για εφαρμογές για τη δημιουργία, επικύρωση και ενίσχυση υπογραφών³. Το πρότυπο καλύπτει τις νομικές απαιτήσεις πολιτικής, τις απαιτήσεις ασφάλειας πληροφοριών (σύστημα διαχείρισης), τη δημιουργία υπογραφών, τις απαιτήσεις διαδικασιών επικύρωσης και διεύρυνσης υπογραφών, απαιτήσεις πολιτικής ανάπτυξης και καδικοποίησης, καθώς και πρόσθετες γενικές απαιτήσεις. Τα Protection Profiles (PPs) που απαιτούνται για την αξιολόγηση κατά Common Criteria εφαρμογών δημιουργίας υπογραφών και εφαρμογών επικύρωσης υπογραφών ορίζονται στο πρότυπο CEN EN 419 111 [13] ως «προφίλ προστασίας για εφαρμογές δημιουργίας και επικύρωσης υπογραφών».

Ένα σημαντικό εργαλείο για τα βασιζόμενα μέρη είναι επίσης η σειρά ETSI TS 119 172 για τις πολιτικές υπογραφής. Αυτή η σειρά προτύπων επιτρέπει τον ορισμό και τις προδιαγραφές των κανόνων που πρέπει να εφαρμόζονται κατά τη δημιουργία, ενίσχυση ή/και επικύρωση των υπογραφών.

7.2.2.3 Εφαρμογές Δημιουργίας Υπογραφών

Το κανονιστικό πλαίσιο δεν έχει συγκεκριμένες απαιτήσεις για εφαρμογές δημιουργίας υπογραφών. Ως εκ τούτου υπάρχουν διάφοροι τρόποι υλοποίησης της διαδικασίας δημιουργίας υπογραφής.

Η διαδικασία δημιουργίας υπογραφής μπορεί να εκτελεστεί εξ ολοκλήρου στο περιβάλλον του υπογράφοντος. Ο υπογράφων διατηρεί τη συσκευή δημιουργίας υπογραφής του, π.χ. έξυπνη κάρτα, και υπογράφει με μια εφαρμογή που βρίσκεται στον υπολογιστή του. Ο υπογράφων μπορεί επίσης να υπογράψει χρησιμοποιώντας την εγκεκριμένη διάταξη δημιουργίας υπογραφής του (π.χ. ηλεκτρονική ταυτότητα (eID)) για να δημιουργήσει μια εγκεκριμένη υπογραφή σε δεδομένα (π.χ. ένα PDF, μια φόρμα, κτλ.) που έχει ετοιμάσει ένας διακομιστής.

Εναλλακτικά, ο υπογράφων μπορεί να βασίζεται σε συσκευές και υλοποιήσεις απομακρυσμένης δημιουργίας υπογραφών (remote signatures). Σε αυτή την περίπτωση, ένα περισσότερο ή λιγότερο σημαντικό κομμάτι της διαδικασίας δημιουργίας υπογραφής, είναι δυνατό να ανατεθεί σε έναν ΠΥΕ. Η αυστηρή απαίτηση

³ Η ενίσχυση των υπογραφών είναι η διαδικασία με την οποία ορισμένο υλικό (π.χ. χρονοσφραγίδες, δεδομένα επικύρωσης και ακόμη και υλικό που σχετίζεται με την αρχειοθέτηση) ενσωματώνεται στις υπογραφές για να τις κάνει πιο ανθεκτικές στις αλλαγές ή για να επεκτείνει τη διάρκειά τους.

είναι ο υπογράφων να μπορεί να ελέγχει την ενεργοποίηση του ιδιωτικού κλειδιού. Για το σκοπό αυτό, ο υπογράφων πρέπει να έχει τον αποκλειστικό έλεγχο του ιδιωτικού κλειδιού του. Όλα τα υπόλοιπα, ακόμη και η διαχείριση του ιδιωτικού κλειδιού (παραγωγή, αποθήκευση), μπορούν να ανατεθούν σε έναν ΠΥΕ και να υλοποιηθούν μέσω ενός ασφαλούς αποθετηρίου ιδιωτικού κλειδιού, π.χ. HSM. Η λύση των απομακρυσμένων υπογραφών έχει ιδιαίτερο ενδιαφέρον από την άποψη της φιλικότητας προς τον χρήστη: η κινητικότητα αυξάνεται καθώς ο έλεγχος ταυτότητας του χρήστη μπορεί να περιοριστεί σε λίγα στοιχεία (π.χ. ένα κινητό τηλέφωνο).

Ωστόσο, ο υπογράφων πρέπει να εμπιστεύεται τον ΠΥΕ για την ισχυρή προστασία του ιδιωτικού κλειδιού όταν αυτό το διαχειρίζεται ο τελευταίος. Χάρη στον κανονισμό eIDAS, όσον αφορά τις εγκεκριμένες ηλεκτρονικές υπογραφές, η συσκευή δημιουργίας υπογραφής πρέπει να είναι πιστοποιημένη (QSCD) και ο ΠΥΕ που διαχειρίζεται ιδιωτικά κλειδιά πρέπει να είναι κατάλληλος (ΕΠΥΕ) και να εποπτεύεται ως μέρος της πιστοποίησης QSCD.

Βιβλιοθήκες Ανοικτού Κώδικα: Η ΕΕ χρηματοδότησε την ανάπτυξη των εργαλείων Digital Signature Service (DSS), ως μέρος των δομικών στοιχείων (building blocks⁴) για το eSignature, που διατίθεται από το Join-up⁵ για τη δημιουργία και την επικύρωση εγκεκριμένων ηλεκτρονικών υπογραφών.

7.2.3 Ηλεκτρονικές Σφραγίδες

Οι ηλεκτρονικές σφραγίδες (electronic seals) είναι μια άλλη υπηρεσία εμπιστοσύνης που ορίζεται από το eIDAS που επιτρέπει σε νομικά πρόσωπα, όπως εταιρείες και άλλους οργανισμούς, να υπογράφουν ηλεκτρονικά έγγραφα και να τα πιστοποιούν ως γνήσια. Πρόκειται για δεδομένα σε ηλεκτρονική μορφή, τα οποία επισυνάπτονται ή συνδέονται λογικά με άλλα δεδομένα σε ηλεκτρονική μορφή για να διασφαλιστεί η προέλευση και η ακεραιότητα των τελευταίων, όπου ο δημιουργός της σφραγίδας είναι ένα νομικό πρόσωπο (σε αντίθεση με την ηλεκτρονική υπογραφή που εκδίδεται από ένα φυσικό πρόσωπο).

Με αυτόν τον τρόπο, οι ηλεκτρονικές σφραγίδες μπορεί να λειτουργήσουν ως αποδεικτικά στοιχεία ότι ένα ηλεκτρονικό έγγραφο εκδόθηκε από ένα νομικό πρόσωπο, διασφαλίζοντας την βεβαιότητα της προέλευσης και της ακεραιότητας του εγγράφου. Παρ' όλα αυτά, σε όλη την Ευρωπαϊκή Ένωση, όταν μία συναλλαγή απαιτεί μια εγκεκριμένη ηλεκτρονική σφραγίδα από ένα νομικό πρόσωπο, μια εγκεκριμένη ηλεκτρονική υπογραφή από τον εξουσιοδοτημένο εκπρόσωπο του νομικού προσώπου είναι εξίσου αποδεκτή.

'Όπως με τις ηλεκτρονικές υπογραφές, υπάρχουν προηγμένες και εγκεκριμένες ηλεκτρονικές σφραγίδες που προσφέρουν πρόσθετα οφέλη σε βασικές ηλεκτρονικές σφραγίδες.

7.2.4 Ηλεκτρονικές Χρονοσφραγίδες

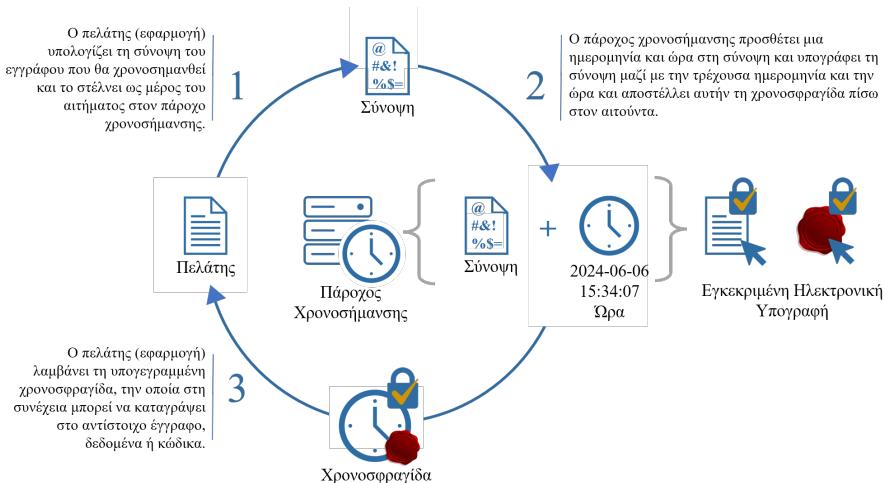
Οι προηγμένες ηλεκτρονικές υπογραφές και σφραγίδες δεν παρέχουν καμία απόδειξη σχετικά με το χρόνο δημιουργίας της υπογραφής ή της σφραγίδας. Μια τέτοια αδιαμφισβήτητη απόδειξη παρέχεται μόνο από μια αξιόπιστη χρονοσφραγίδα που προστίθεται στα υπογεγραμμένα δεδομένα. Μια ηλεκτρονική χρονοσφραγίδα είναι δεδομένα σε ηλεκτρονική μορφή που συσχετίζουν άλλα δεδομένα σε ηλεκτρονική μορφή με έναν συγκεκριμένο χρόνο που αποδεικνύει ότι τα τελευταία δεδομένα υπήρχαν εκείνη τη δεδομένη χρονική στιγμή. Οι ηλεκτρονικές χρονοσφραγίδες παρέχουν τα μέσα για τον καθορισμό του χρόνου υπογραφής ή έκδοσης ενός εγγράφου ή δήλωσης ή του χρόνου σύναψης σύμβασης. Χρησιμοποιούνται επίσης για τη μακροπρόθεσμη εγκυρότητα και διατήρηση ηλεκτρονικών αρχείων και για συμβολαιογραφικές πράξεις.

Η δημιουργία μιας ηλεκτρονικής χρονοσφραγίδας αποτελείται από τα ακόλουθα βήματα, όπως απεικονίζονται και στο Σχήμα 7.6.

⁴Τα δομικά στοιχεία (building blocks) αποτελούν μια ανοιχτή και επαναχρησιμοποιήσιμη ψηφιακή λύση. Μπορεί να έχει τη μορφή ενός πλαισίου, ενός προτύπου, ενός λογισμικού ή ενός λογισμικού ως υπηρεσίας (Software as a Service – SaaS) ή οποιουδήποτε συνδυασμού τους.

⁵<https://joinup.ec.europa.eu/collection/digital-signature-service>

- Ο πελάτης στέλνει σε έναν πάροχο υπηρεσιών ηλεκτρονικών χρονοσφραγίδων της επιλογής του, ένα αίτημα που περιέχει μια σήμανση των δεδομένων για τα οποία ο χρήστης θέλει να έχει χρονική σήμανση.
- Ο πάροχος υπηρεσιών ηλεκτρονικών χρονοσφραγίδων προσαρτά στη σύνοψη την ημερομηνία και ώρα που λαμβάνει χώρα η συναλλαγή, υπογράφει τα δεδομένα και τα στέλνει πίσω στον πελάτη.
- Έχοντας λάβει τη χρονοσφραγίδα, ο πελάτης μπορεί να την επισυνάψει στα δεδομένα.



Σχήμα 7.6: Διαδικασία δημιουργίας χρονοσφραγίδας.

7.2.5 Ηλεκτρονικές Υπηρεσίες Συστημένης Παράδοσης

Η ηλεκτρονική υπηρεσία συστημένης παράδοσης (Electronic Registered Delivery Service) είναι μια υπηρεσία που καθιστά δυνατή τη μετάδοση δεδομένων μεταξύ τρίτων με ηλεκτρονικά μέσα και παρέχει αποδεικτικά στοιχεία σχετικά με το χειρισμό των μεταδιδόμενων δεδομένων, συμπεριλαμβανομένης της απόδειξης αποστολής και λήψης των δεδομένων. Είναι η υπηρεσία που μπορεί να παρέχει μη αποποίηση παραλαβής, η οποία είναι απαραίτητη σε πολλές ηλεκτρονικές υπηρεσίες, όπως στην ανταλλαγή εγγράφων.

Η ηλεκτρονική υπηρεσία συστημένης παράδοσης ορίζεται από τον κανονισμό ως «υπηρεσία η οποία καθιστά δυνατή τη διαβίβαση δεδομένων μεταξύ τρίτων μερών με ηλεκτρονικά μέσα και παρέχει τεκμήρια σχετικά με τον χειρισμό των διαβιβαζόμενων δεδομένων, περιλαμβανομένης απόδειξης της αποστολής και της παραλαβής των δεδομένων, και η οποία προστατεύει τα διαβιβαζόμενα δεδομένα από τον κίνδυνο απώλειας, κλοπής, βλάβης ή τροποποίησης τους χωρίς άδεια». Στην πράξη, τα δεδομένα που αναφέρονται από αυτόν τον ορισμό ότι διαβιβάζονται στο πλαίσιο μιας τέτοιας υπηρεσίας ηλεκτρονικής παράδοσης από έναν αποστολέα σε έναν παραλήπτη, μπορεί να είναι οποιουδήποτε τύπου, δομημένα ή μη, συμπεριλαμβανομένων ηλεκτρονικών αρχείων. Τα μέσα μετάδοσης μπορούν επίσης να είναι οποιουδήποτε είδους, συμπεριλαμβανομένου του ηλεκτρονικού ταχυδρομείου.

Το eDelivery μπορεί να χρησιμοποιηθεί από κάθε είδους οντότητες που επιθυμούν να μοιράζονται ηλεκτρονικά αρχεία με ασφάλεια, όπως δημόσιες διοικήσεις, οργανισμοί, επιχειρήσεις και πολίτες.

Οι εγκεκριμένες ηλεκτρονικές υπηρεσίες συστημένης παράδοσης, παρέχονται από έναν ή περισσότερους εγκεκριμένους παρόχους υπηρεσιών εμπιστοσύνης, εξασφαλίζοντας την ταυτοποίηση του αποστολέα αλλά και του παραλήπτη πριν από την παράδοση των δεδομένων. Συγκεκριμένα, οι εγκεκριμένες ηλεκτρονικές υπηρεσίες συστημένης παράδοσης πληρούν τις ακόλουθες απαιτήσεις:

- Παρέχονται από έναν ή περισσότερους εγκεκριμένους παρόχους υπηρεσιών εμπιστοσύνης.

- Εξασφαλίζουν με υψηλό επίπεδο εμπιστοσύνης την ταυτοποίηση του αποστολέα.
- Εξασφαλίζουν την ταυτοποίηση του αποδέκτη πριν από την παράδοση των δεδομένων.
- Η αποστολή και η λήψη των δεδομένων διασφαλίζονται με προηγμένη ηλεκτρονική υπογραφή ή προηγμένη ηλεκτρονική σφραγίδα εγκεκριμένου παρόχου υπηρεσιών εμπιστοσύνης κατά τρόπο που να αποκλείει τη δυνατότητα μη ανιχνεύσιμης τροποποίησης των δεδομένων.
- Οποιαδήποτε τροποποίηση των δεδομένων που απαιτούνται για τους σκοπούς της αποστολής ή της λήψης των δεδομένων δηλώνεται σαφώς στον αποστολέα και στον αποδέκτη των δεδομένων.
- Η ημερομηνία και ο χρόνος αποστολής, παραλαβής και οποιαδήποτε αλλαγή των στοιχείων αναφέρεται με εγκεκριμένη ηλεκτρονική χρονοσφραγίδα.

Η αποστολή και η λήψη των δεδομένων διασφαλίζονται με προηγμένη ηλεκτρονική υπογραφή ή προηγμένη ηλεκτρονική σφραγίδα εγκεκριμένου παρόχου υπηρεσιών εμπιστοσύνης, κατά τρόπο που να αποκλείει τη δυνατότητα μη ανιχνεύσιμης τροποποίησης των δεδομένων, ενώ η ημερομηνία και ο χρόνος αποστολής, παραλαβής και οποιαδήποτε αλλαγή των στοιχείων αναφέρεται με εγκεκριμένη ηλεκτρονική χρονοσφραγίδα.

7.2.6 Πιστοποίηση Γνησιότητας Ιστοτόπων

Οι υπηρεσίες επαλήθευσης της ταυτότητας ιστοτόπων αποτελούν ένα μέσο με το οποίο ένας επισκέπτης του ιστότοπου μπορεί να βεβαιωθεί ότι υπάρχει πραγματική και νόμιμη οντότητα πίσω από τον ιστότοπο. Οι υπηρεσίες αυτές συμβάλλουν στη δημιουργία σχέσεων εμπιστοσύνης μεταξύ χρηστών και παρόχων διαδικτυακών υπηρεσιών. Οι ΠΥΕ μπορούν να εκδίδουν και διαχειρίζονται τους ακόλουθους τύπους πιστοποιητικών για τους ιστότοπους:

- Πιστοποιητικό γνησιότητας ιστότοπου: βεβαίωση η οποία επιτρέπει την επαλήθευση της γνησιότητας ενός ιστότοπου και συνδέει τον ιστότοπο με το φυσικό ή νομικό πρόσωπο στο οποίο έχει εκδοθεί το πιστοποιητικό.
- Εγκεκριμένο πιστοποιητικό γνησιότητας ιστότοπου: πιστοποιητικό για την επαλήθευση της γνησιότητας ιστότοπου που εκδίδεται από εγκεκριμένο πάροχο υπηρεσιών εμπιστοσύνης και περιέχει τα ακόλουθα:
 - Ένδειξη, τουλάχιστον σε μορφή κατάλληλη για την αυτοματοποιημένη επεξεργασία, ότι το πιστοποιητικό έχει εκδοθεί ως εγκεκριμένο πιστοποιητικό γνησιότητας ιστότοπου.
 - Ένα σύνολο δεδομένων που αντιπροσωπεύουν αδιαμφισβήτητα τον εγκεκριμένο πάροχο υπηρεσιών εμπιστοσύνης ο οποίος έχει εκδώσει τα εγκεκριμένα πιστοποιητικά και συμπεριλαμβάνουν τουλάχιστον το κράτος-μέλος στο οποίο είναι εγκατεστημένος και (α) σε περίπτωση που πρόκειται για νομικό πρόσωπο: το όνομα και, κατά περίπτωση, τον αριθμό μητρώου του, όπως αναφέρεται στα επίσημα αρχεία, (β) σε περίπτωση που πρόκειται για φυσικό πρόσωπο: το όνομα του προσώπου.
 - Για τα φυσικά πρόσωπα: τουλάχιστον το όνομα του προσώπου για το οποίο έχει εκδοθεί το πιστοποιητικό ή ψευδώνυμο. Εάν χρησιμοποιείται ψευδώνυμο, πρέπει να αναφέρεται σαφώς – για τα νομικά πρόσωπα: τουλάχιστον το όνομα του νομικού προσώπου για το οποίο έχει εκδοθεί το πιστοποιητικό και, κατά περίπτωση, τον αριθμό μητρώου του, όπως αναφέρεται στα επίσημα αρχεία.
 - Στοιχεία της διεύθυνσης, συμπεριλαμβανομένης τουλάχιστον της πόλης και του κράτους μέλους, του φυσικού ή νομικού προσώπου για το οποίο έχει εκδοθεί το πιστοποιητικό και, κατά περίπτωση, όπως αναφέρεται στα επίσημα αρχεία.

- Το όνομα χώρου που ανήκει στο φυσικό ή νομικό πρόσωπο για το οποίο έχει εκδοθεί το πιστοποιητικό.
- Λεπτομέρειες για την έναρξη και τη λήξη της περιόδου ισχύος του πιστοποιητικού.
- Τον κωδικό ταυτότητας του πιστοποιητικού, ο οποίος πρέπει να είναι μοναδικός για τον εγκεκριμένο πάροχο υπηρεσιών εμπιστοσύνης.
- Την προηγμένη ηλεκτρονική υπογραφή ή την προηγμένη ηλεκτρονική σφραγίδα του εκδίδοντος εγκεκριμένου παρόχου υπηρεσιών εμπιστοσύνης.
- Την τοποθεσία όπου διατίθεται δωρεάν το πιστοποιητικό το οποίο τεκμηριώνει την προηγμένη ηλεκτρονική υπογραφή ή την προηγμένη ηλεκτρονική σφραγίδα που αναφέρεται στο προηγούμενο στοιχείο.
- Την τοποθεσία των υπηρεσιών κατάστασης ισχύος πιστοποιητικών που παρέχουν πληροφορίες σχετικά με την κατάσταση ισχύος του εγκεκριμένου πιστοποιητικού.

7.2.7 Κατάλογοι Εμπιστοσύνης της ΕΕ

Για να διευκολύνει τη χρήση αξιόπιστων υπηρεσιών, η ΕΕ έχει θεσπίσει έναν μηχανισμό μέσω του οποίου οι χρήστες μπορούν να ενημερώνονται εύκολα για όλους τους παρόχους υπηρεσιών εμπιστοσύνης που είναι εγκατεστημένοι σε ένα κράτος μέλος της ΕΕ και για τους τύπους υπηρεσιών που παρέχουν. Οι αρμόδιες αρχές σε κάθε κράτος μέλος είναι υπεύθυνες για τη διατήρηση ενός καταλόγου παρόχων υπηρεσιών, συμπεριλαμβανομένων των εγκεκριμένων παρόχων, και των υπηρεσιών που παρέχουν αυτοί στο αντίστοιχο κράτος μέλος. Αυτοί οι κατάλογοι δημοσιεύονται με ασφαλή τρόπο, καθώς υπογράφονται ή σφραγίζονται ηλεκτρονικά, και είναι σε μορφή κατάλληλη για αυτόματη επεξεργασία, έτσι ώστε να μπορούν να ενσωματωθούν εύκολα σε εφαρμογές⁶.

Οι κατάλογοι εμπιστοσύνης (trusted lists) είναι βασικά στοιχεία για την οικοδόμηση εμπιστοσύνης μεταξύ των φορέων εκμετάλλευσης ηλεκτρονικών αγορών, επιτρέποντας στους χρήστες να προσδιορίζουν την κατάλληλη κατάσταση και το ιστορικό κατάστασης των παρόχων υπηρεσιών εμπιστοσύνης και των υπηρεσιών τους. Τα κράτη μέλη μπορούν να περιλαμβάνουν στους καταλόγους πληροφορίες για μη εγκεκριμένους παρόχους υπηρεσιών εμπιστοσύνης, μαζί με πληροφορίες που σχετίζονται με τις μη εγκεκριμένες υπηρεσίες εμπιστοσύνης που παρέχονται από αυτούς. Πρέπει να αναφέρεται σαφώς ότι δεν πληρούν τις προϋποθέσεις σύμφωνα με τον κανονισμό (ΕΕ) αριθ. 910/2014.

7.3 Ψηφιακά Πορτοφόλια

Τα τελευταία χρόνια έχουν κάνει δυναμικά την εμφάνιση τους οι λύσεις ψηφιακών πορτοφολιών για τη διαχείριση ψηφιακών ταυτοτήτων (digital identity)⁷. Στο πλαίσιο αυτό η Ευρωπαϊκή Ένωση προωθεί τη χρήση ενός Ευρωπαϊκού Πορτοφολιού Ψηφιακής Ταυτότητας. Ο πρωταρχικός στόχος του προτεινόμενου Ευρωπαϊκού Πορτοφολιού Ψηφιακής Ταυτότητας είναι να προωθήσει αξιόπιστες ψηφιακές ταυτότητες για όλους τους Ευρωπαίους, επιτρέποντας στους χρήστες να έχουν τον έλεγχο των δικών τους διαδικτυακών αλληλεπιδράσεων και παρουσίας στο διαδίκτυο.

Το ψηφιακό πορτοφόλι μπορεί να θεωρηθεί ως ένας συνδυασμός πολλών προϊόντων και υπηρεσιών εμπιστοσύνης που επιτρέπει στους χρήστες (φυσικά και νομικά πρόσωπα) να ζητούν, να λαμβάνουν, να αποθηκεύουν και να μοιράζονται με ασφάλεια τα συσχετίζομενα δεδομένα τους, επιτρέποντάς τους να έχουν πρόσβαση σε διαδικτυακές υπηρεσίες και να υπογράφουν/σφραγίζουν ηλεκτρονικά έγγραφα.

⁶<https://eidas.ec.europa.eu/efda/tl-browser/>

⁷Η ψηφιακή ταυτότητα ορίζεται ως η μοναδική αντιπροσώπευση ενός υποκειμένου που συμμετέχει σε μια διαδικτυακή συναλλαγή [14]. Μια ψηφιακή ταυτότητα είναι πάντα μοναδική στο πλαίσιο μιας ψηφιακής υπηρεσίας, αλλά δεν είναι απαραίτητο να αναγνωρίζει μοναδικά το υποκειμένο σε όλα τα πλαίσια. Με άλλα λόγια, η πρόσβαση σε μια ψηφιακή υπηρεσία δεν σημαίνει ότι η πραγματική ταυτότητα του υποκειμένου είναι γνωστή.

Υπάρχει πληθώρα περιπτώσεων χρήσης των ψηφιακών πορτοφολιών, κάποιες από τις οποίες είναι:

- Ασφαλής και αξιόπιστη ταυτότητα για πρόσβαση σε διαδικτυακές υπηρεσίες:** Ο ασφαλής έλεγχος ταυτότητας είναι μια λειτουργία του πορτοφολιού όπου τα βασιζόμενη μέρη (relying parties) προσδιορίζουν τους χρήστες με ένα καθορισμένο σύνολο δεδομένων ταυτοποίησης για τους σκοπούς της δυνατότητας πρόσβασης σε διαδικτυακές δημόσιες και ιδιωτικές υπηρεσίες.
- Κινητικότητα και ψηφιακή άδεια οδήγησης:** Το πορτοφόλι μπορεί να χρησιμοποιηθεί ως μια πλήρως ψηφιοποιημένη άδεια οδήγησης για διαδικτυακές και εκτός σύνδεσης χρήσεις. Θα μπορούσε να συνδέεται με μια σειρά περαιτέρω βεβαιώσεων που προσφέρονται από δημόσιους ή ιδιωτικούς παρόχους που καλύπτουν νομικές απαιτήσεις (π.χ. Πιστοποιητικό Επαγγελματικής Ικανότητας) ή επιχειρηματικές απαιτήσεις και πρότυπα (π.χ. για διόδια) στον τομέα των οδικών μεταφορών.
- Υγεία:** Η εύκολη πρόσβαση στα δεδομένα υγείας είναι ζωτικής σημασίας τόσο σε εθνικό όσο και σε διασυνοριακό πλαίσιο. Με βάση την εμπειρία από το ψηφιακό πιστοποιητικό COVID-19 της ΕΕ, το ψηφιακό πορτοφόλι μπορεί να επιτρέψει την πρόσβαση σε φακέλους και ιστορικό ασθενών, ηλεκτρονικές συνταγές κ.α.
- Εκπαίδευση / Δίπλωμα:** Η παροχή εγγράφων για τις διαδικασίες αναγνώρισης προσόντων μπορεί να είναι δαπανηρή και χρονοβόρα για τους τελικούς χρήστες, τις εταιρείες και τους εργοδότες, τους παρόχους εκπαίδευσης και κατάρτισης και άλλα ακαδημαϊκά ίδρυματα. Για παράδειγμα, οι ψηφιακές βεβαιώσεις διπλωμάτων θα μπορούσαν να κοινοποιηθούν διασυνοριακά σε επαλήθευσιμη και αξιόπιστη μορφή σε άλλο ίδρυμα εκπαίδευσης ή κατάρτισης ή σε μελλοντικό εργοδότη.

7.3.1 Ευρωπαϊκό Ψηφιακό Πορτοφόλι

Ο νέος κανονισμός eIDAS [15] ορίζει το «ευρωπαϊκό πορτοφόλι ψηφιακής ταυτότητας» ως ένα «προϊόν και υπηρεσία που επιτρέπει στον χρήστη να αποθηκεύει δεδομένα ταυτότητας, διαπιστευτήρια και χαρακτηριστικά που συνδέονται με την ταυτότητά του, να τα παρέχει σε βασιζόμενα μέρη κατόπιν αιτήματος και να τα χρησιμοποιεί για επαλήθευση ταυτότητας σε επιγραμμικές (online) και μη επιγραμμικές υπηρεσίες και να δημιουργεί εγκεκριμένες ηλεκτρονικές υπογραφές και σφραγίδες».

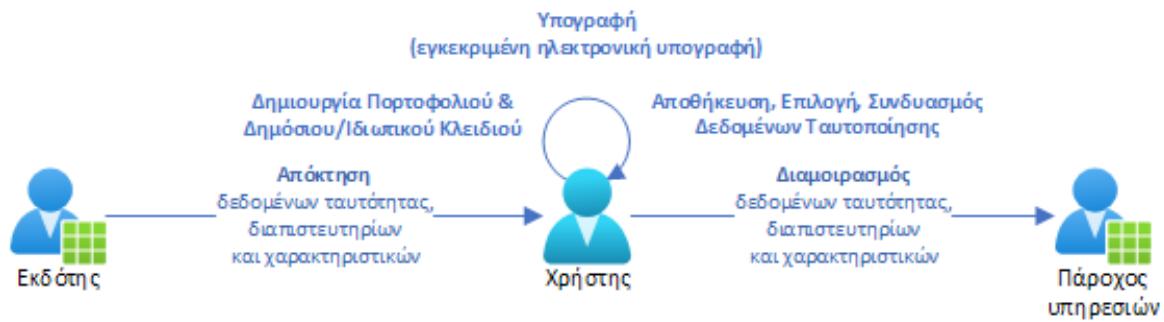
Τα ευρωπαϊκά πορτοφόλια ψηφιακής ταυτότητας εκδίδονται από κράτος-μέλος, με εντολή κράτους-μέλους, ή ανεξάρτητα, ωστόσο αναγνωρίζονται από κράτος μέλος και παρέχονται στον χρήστη τη δυνατότητα:

- Να ζητά με ασφάλεια και να λαμβάνει, να αποθηκεύει, να επιλέγει, να συνδυάζει και να κοινοποιεί, κατά τρόπο διαφανή και ανιχνεύσιμο από τον χρήστη, τα απαραίτητα νομικά δεδομένα ταυτοποίησης προσώπου και την ηλεκτρονική βεβαίωση χαρακτηριστικών για την επαλήθευση της ταυτότητας εντός και εκτός διαδικτύου με σκοπό τη χρήση επιγραμμικών δημόσιων και ιδιωτικών υπηρεσιών.
- Να υπογράφει με εγκεκριμένες ηλεκτρονικές υπογραφές.

Η βασική λειτουργικότητα ενός ψηφιακού πορτοφολιού αποτυπώνεται στο Σχήμα 7.7. Ο χρήστης, ως κάτοχος ενός ψηφιακού πορτοφολιού, έχει τη δυνατότητα να αιτηθεί και να λάβει από έναν εκδότη, δεδομένα ταυτοποίησης, βεβαιώσεις χαρακτηριστικών, και άλλα διαπιστευτήρια, να τα αποθηκεύει και να τα διαχειρίζεται κατάλληλα με τη χρήση του ψηφιακού πορτοφολιού, καθώς επίσης και να τα κοινοποιεί σε τρίτα βασιζόμενα μέρη για την πρόσβαση σε υπηρεσίες.

Για τις ανάγκες αυτές, τα πορτοφόλια ψηφιακής ταυτότητας παρέχουν κοινή διεπαφή (interface):

- Σε εγκεκριμένους και μη εγκεκριμένους παρόχους υπηρεσιών εμπιστοσύνης που εκδίδουν εγκεκριμένες και μη εγκεκριμένες ηλεκτρονικές βεβαιώσεις χαρακτηριστικών ή άλλα εγκεκριμένα και μη εγκεκριμένα πιστοποιητικά με σκοπό την έκδοση των εν λόγω βεβαιώσεων και πιστοποιητικών για το ευρωπαϊκό πορτοφόλι ψηφιακής ταυτότητας.



Σχήμα 7.7: Βασική λειτουργικότητα ψηφιακού πορτοφολιού.

2. Όστε τα βασιζόμενα μέρη να ζητούν και να επικυρώνουν δεδομένα ταυτοποίησης προσώπων και ηλεκτρονικές βεβαιώσεις χαρακτηριστικών.
3. Για την προσκόμιση, στα βασιζόμενα μέρη, δεδομένων ταυτοποίησης προσώπου, ηλεκτρονικής βεβαιώσης χαρακτηριστικών ή άλλων δεδομένων, όπως διαπιστευτηρίων, τοπικά, χωρίς να απαιτείται για το πορτοφόλι πρόσβαση στο διαδίκτυο.

Βιβλιογραφία

- [1] D. Cooper et al. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. en. Tech. rep. RFC5280. RFC Editor, May 2008, RFC5280. doi: 10 . 17487 / rfc5280.
- [2] S. Chokhani, W. Ford, R. Sabett, C. Merrill, and S. Wu. *Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework*. RFC 3647. RFC Editor, Nov. 2003.
- [3] ITU. *X.509: Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks*. Feb. 2011. URL: <http://www.itu.int/rec/T-REC-X.509>.
- [4] S. Santesson et al. *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. en. Tech. rep. RFC6960. RFC Editor, June 2013, RFC6960. doi: 10.17487/rfc6960.
- [5] Elaine B. Barker and Quynh H. Dang. *Recommendation for Key Management Part 3: Application-Specific Key Management Guidance*. Tech. rep. Jan. 2015. doi: 10.6028/nist.sp.800-57pt3r1.
- [6] European Parliament and Council. “Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC”. In: *Official Journal of the European Union* (2014). URL: <http://data.europa.eu/eli/reg/2014/910/oj>.
- [7] *Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for signature creation and validation*. Tech. rep. TR 119 100. European Telecommunications Standards Institute (ETSI), 2016. URL: https://www.etsi.org/deliver/etsi_tr/119100_119199/119100/01.01.01_60/tr_119100v010101p.pdf.
- [8] *Electronic Signatures and Infrastructures (ESI); XAdES Baseline Profile*. Version V2.1.1. European Telecommunications Standards Institute (ETSI), 2012. URL: https://www.etsi.org/deliver/etsi_ts/103100_103199/103171/02.01.01_60/ts_103171v020101p.pdf.

- [9] *Electronic Signatures and Infrastructures (ESI); PADES Baseline Profile*. Version V2.2.2. European Telecommunications Standards Institute (ETSI), 2013. URL: https://www.etsi.org/deliver/etsi_ts/103100_103199/103172/02.02.02_60/ts_103172v020202p.pdf.
- [10] *Electronic Signatures and Infrastructures (ESI); CADES Baseline Profile*. Version V2.2.1. European Telecommunications Standards Institute (ETSI), 2013. URL: https://www.etsi.org/deliver/etsi_ts/103100_103199/103173/02.01.01_60/ts_103173v020101p.pdf.
- [11] *Electronic Signatures and Infrastructures (ESI); ASiC Baseline Profile*. Version V2.2.1. European Telecommunications Standards Institute (ETSI), 2013. URL: https://www.etsi.org/deliver/etsi_ts/103100_103199/103174/02.02.01_60/ts_103174v020201p.pdf.
- [12] *Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 1: General requirements*. Version V1.1.1. European Telecommunications Standards Institute (ETSI), 2016. URL: https://www.etsi.org/deliver/etsi_ts/119100_119199/119101/01.01_60/ts_119101v010101p.pdf.
- [13] *Protection profiles for secure signature creation device - Part 1: Overview*. European Committee for Standardization (CEN), 2015. URL: <https://www.en-standard.eu/csn-en-419211-1-protection-profiles-for-secure-signature-creation-device-part-1-overview/>.
- [14] Paul A Grassi, Michael E Garcia, and James L Fenton. *Digital identity guidelines: revision 3*. Tech. rep. NIST SP 800-63-3. Gaithersburg, MD: National Institute of Standards and Technology, June 2017, NIST SP 800-63-3. DOI: 10.6028/NIST.SP.800-63-3.
- [15] European Parliament and Council. “Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL amending Regulation (EU) No 910/2014 as regards establishing a framework for a European Digital Identity, COM(2021) 281”. In: *Official Journal of the European Union* (2021). URL: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0281>.

ΚΕΦΑΛΑΙΟ 8

ΜΗΧΑΝΙΣΜΟΙ ΕΝΙΣΧΥΣΗΣ ΤΟΥ ΑΠΟΡΡΗΤΟΥ

Περίληψη

Στο κεφάλαιο αυτό παρουσιάζονται διάφοροι κρυπτογραφικοί μηχανισμοί ενίσχυσης του απορρήτου, που αποτελούν προαπαιτούμενα για την ανάπτυξη κρυπτογραφικών εφαρμογών που έχουν ως απώτερο στόχο την προστασία της ιδιωτικότητας των χρηστών αλλά και την διασφάλιση του εμπορικού/εταιρικού απορρήτου (βλέπε Κεφάλαιο 12). Πιο αναλυτικά, στην Ενότητα 8.1 γίνεται αναφορά σε ασφαλείς υπολογισμούς πολλαπλών οντοτήτων (MPC), παραθέτοντας ως παράδειγμα το πρωτόκολλο του Yao και κάνοντας αναφορά σε άλλα θεμελιώδη πρωτόκολλα. Στην Ενότητα 8.2 παρουσιάζεται με αρκετές λεπτομέρειες η ομοιορφική κρυπτογράφηση και παρατίθενται τόσο η μερική όσο και η πλήρης ομοιορφική κρυπτογράφηση. Στην Ενότητα 8.3 περιγράφονται οι αποδείξεις μηδενικής γνώσης (ZKP) και αναφέρονται ενδεικτικά δύο παραδείγματα τέτοιων αποδείξεων: της απόδειξης γνώσης του διακριτού λογάριθμου και της απόδειξης ισότητας δύο διακριτών λογάριθμων. Στην Ενότητα 8.4 γίνεται αναφορά σε τρία σχήματα υπογραφών ενίσχυσης του απορρήτου και συγκεκριμένα στις τυφλές υπογραφές, στις ομαδικές υπογραφές και στις υπογραφές δακτυλίου. Στην Ενότητα 8.5 παρουσιάζονται συνοπτικά η κρυπτογράφηση βάσει ταυτότητα (IBE) και η κρυπτογράφηση βάσει χαρακτηριστικών (ABE), ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών αρχών της συμμετρικής (Κεφάλαιο 1) και ασύμμετρης (Κεφάλαιο 2) κρυπτογραφίας, καθώς και των συναρτήσεων σύνοψης (Κεφάλαιο 3).

8.1 Ασφαλείς Υπολογισμοί Πολλαπλών Οντοτήτων

Ο ασφαλής υπολογισμός πολλαπλών οντοτήτων (Secure Multi-Party Computation – MPC) επιτρέπει σε μια ομάδα να εκτελέσει από κοινού έναν υπολογισμό χωρίς να αποκαλύπτει τα ιδιωτικά δεδομένα (private data) εισόδου των συμμετεχόντων [1]. Οι συμμετέχοντες συμφωνούν αρχικά σε μια συνάρτηση υπολογισμού και στη συνέχεια μπορούν να χρησιμοποιήσουν ένα πρωτόκολλο MPC για να υπολογίσουν από κοινού την έξοδο αυτής της συνάρτησης έχοντας ως είσοδο ιδιωτικά δεδομένα τα οποία ίμως δεν θα αποκαλυφθούν. Ύστερα από την πρώτη εμφάνιση του από τον Yao το 1982 [2], ο υπολογισμός πολλαπλών οντοτήτων έχει αναπτυχθεί

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx-978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

τόσο θεωρητικά όσο και πρακτικά ως ένα σημαντικό εργαλείο για τη ανάπτυξη εφαρμογών διασφάλισης του απορρήτου και της ιδιωτικότητας.

Ο Yao [2] εισήγαγε ουσιαστικά μια γενική έννοια του ασφαλούς υπολογισμού, στην οποία οι συμμετέχοντες θέλουν να υπολογίσουν από κοινού μια συνάρτηση $f(x_1, x_2, \dots, x_m)$, όπου x_i είναι η ιδιωτική είσοδος μιας i οντότητας. Το παράδειγμα που χρησιμοποίησε για την περιγραφή αυτού του ασφαλούς υπολογισμού ήταν το πρόβλημα του εκατομμυριούχου: Η Αλίκη και ο Μπάμπης είναι δύο εκατομμυριούχοι που θέλουν να βρουν ποιος είναι πλουσιότερος χωρίς όμως να αποκαλυφθεί το ακριβές ποσό της περιουσίας τους. Ο Yao σε αυτό το πρόβλημα πρότεινε μια λύση που επιτρέπει στην Αλίκη και στον Μπάμπη να ικανοποιήσουν την περιέργειά τους ενώ ταυτόχρονα σέβονται τον παραπάνω περιορισμό. Αυτό το πρωτόκολλο παραμένει η βάση για πολλές από τις πιο αποτελεσματικές εφαρμογές του MPC. Ένα πρωτόκολλο MPC θεωρείται ασφαλές εάν κανένας συμμετέχων δεν μπορεί να μάθει περισσότερα από την περιγραφή της δημόσιας συνάρτησης υπολογισμού και το αποτέλεσμα υπολογισμού.

Στα επόμενα είκοσι χρόνια που ακολούθησαν μετά τον Yao, ο ασφαλής υπολογισμός πολλαπλών οντοτήτων είχε κυρίως θεωρητικό ενδιαφέρον. Μόνο τη δεκαετία του 2000 οι αλγορίθμικές βελτιώσεις και το υπολογιστικό κόστος έφτασαν σε ένα σημείο όπου έγινε ρεαλιστικό να σκεφτούμε την κατασκευή πρακτικών συστημάτων χρησιμοποιώντας υπολογισμούς πολλαπλών οντοτήτων γενικής χρήσης. Το Fairplay [3] αποτελεί την πρώτη αξιοσημείωτη εφαρμογή γενικού σκοπού ενός ασφαλούς υπολογισμού. Ουσιαστικά, ανέδειξε ότι ένα πρόγραμμα διασφάλισης της ιδιωτικότητας θα μπορούσε να εκφραστεί σε μια γλώσσα υψηλού επιπέδου και να μεταγλωτιστεί σε ένα εκτελέσιμο που θα μπορούσε να εκτελεστεί ως πρωτόκολλο πολλαπλών οντοτήτων από τους συμμετέχοντες που κατέχουν τα ιδιωτικά δεδομένα. Ωστόσο, η επεκτασιμότητα και η απόδοσή του δεν επέτρεψε τη χρήση του σε πραγματικές εφαρμογές. Η μεγαλύτερη εφαρμογή που αναφέρεται στο [3] ήταν ο υπολογισμός του διάμεσου δύο ταξινομημένων πινάκων, όπου η είσοδος κάθε οντότητας ήταν δέκα αριθμοί 16-bit σε ταξινομημένη σειρά, και η εκτέλεσή του έπαιρνε πάνω από 7 δευτερόλεπτα σε 4383 πύλες (και τα δύο οντότητες ήταν διασυνδεδεμένες μέσω δικτύου LAN). Από τότε μέχρι και σήμερα, η ταχύτητα των πρωτοκόλλων MPC έχει βελτιωθεί περισσότερο από πέντε τάξεις μεγέθους λόγω ενός συνδυασμού βελτιώσεων στο υλικό, στις κρυπτογραφικές τεχνικές και στο δίκτυο. Αυτό έχει επιτρέψει στις εφαρμογές MPC να εμφανιστούν σε ένα ευρύ φάσμα ενδιαφέροντων και σημαντικών εφαρμογών, όπως τις ασφαλείς δημοπρασίες, τις ψηφιακές κάλπες και την ασφαλή μηχανική μάθηση.

Στις υποενότητες που ακολουθούν, παρουσιάζεται λεπτομερώς ως παράδειγμα το πρώτο πρωτόκολλο του Yao που έδινε λύση στο πρόβλημα του εκατομμυριούχου και επιπλέον αναφέρονται ενδεικτικά άλλα θεμελιώδη πρωτόκολλα που αποτελούν γενικές προσεγγίσεις για την πραγματοποίηση ασφαλών υπολογισμών ακολουθώντας το μοντέλο ασφάλειας παθητικών αντιπάλων (Ορισμός 8.1).

Ορισμός 8.1 (Παθητικός Αντίπαλος). Ένας παθητικός (passive ή semi-honest) αντίπαλος είναι ένας διεφθαρμένος συμμετέχων που εκτελεί ένα πρωτόκολλο με τιμιότητα, αλλά μπορεί να προσπαθήσει να μάθει όσο το δυνατόν περισσότερο από τα μηνύματα που ανταλλάσσονται μεταξύ των άλλων συμμετεχόντων.

8.1.1 Το Πρωτόκολλο του Yao (Garbled Circuit)

Το πρωτόκολλο του Yao, γνωστό ως Garbled Circuit (GC) [4], αποτελεί ένα κρυπτογραφικό πρωτόκολλο που προτάθηκε από τον Yao για να λύσει το πρόβλημα των εκατομμυριούχων επιτρέποντας έτσι τον ασφαλή υπολογισμό δύο οντοτήτων, στον οποίο οι συμμετέχοντες δεν εμπιστεύονται ο ένας τον άλλο, αλλά μπορούν να υπολογίσουν από κοινού μια συνάρτηση που έχει ως είσοδο τις ιδιωτικές τους τιμές χωρίς την παρουσία μιας αξιόπιστης τρίτης οντότητας.

Το πρωτόκολλο GC αποτελείται από τα ακόλουθα 6 βήματα:

1. Η υποκείμενη συνάρτηση υπολογισμού (όπως το πρόβλημα των εκατομμυριούχων [2]) περιγράφεται ως ένα λογικό δυαδικό κύκλωμα, με πύλες δύο εισόδων, που είναι γνωστό και στα δύο μέρη. Αυτό το βήμα μπορεί να πραγματοποιηθεί προγενέστερα από μια τρίτη οντότητα.

2. Η Αλίκη κρυπτογραφεί (garbles) το κύκλωμα. Η Αλίκη σε αυτό το πρωτόκολλο αποκαλείται ως *garbler*.
3. Η Αλίκη στέλνει το κρυπτογραφημένο κύκλωμα (garbled circuit) στον Μπάμπη μαζί με την κρυπτογραφημένη είσοδο της.
4. Ο Μπάμπης, προκειμένου να υπολογίσει το κύκλωμα, πρέπει επίσης να υπολογίσει τη δική του κρυπτογραφημένη είσοδο. Για το σκοπό αυτό, χρειάζεται την βοήθεια της Αλίκης, γιατί είναι η μόνη που γνωρίζει πώς να κρυπτογραφεί. Ο Μπάμπης μπορεί να κρυπτογραφήσει την είσοδό του με την βοήθεια της Αλίκης κάνοντας χρήση ενός πρωτοκόλλου γνωστού ως μη-συνειδητή μεταφορά (Ορισμός 8.2). Με βάση τον ορισμό αυτού του πρωτοκόλλου, ο Μπάμπης είναι ο παραλήπτης και η Αλίκη ο αποστολέας σε αυτή την μη-συνειδητή μεταφορά.
5. Ο Μπάμπης αποκρυπτογραφεί (evaluates) το κύκλωμα και λαμβάνει τις κρυπτογραφημένες εξόδους. Ο Μπάμπης σε αυτό το πρωτόκολλο αποκαλείται ως *αξιολογητής* (evaluator).
6. Η Αλίκη και ο Μπάμπης αλληλεπιδρούν μεταξύ τους, αποκαλύπτοντας επιμέρους στοιχεία που γνωρίζει ο καθένας, προκειμένου να μάθουν την τελική έξοδο του πρωτοκόλλου.

Ορισμός 8.2 (Μη-συνειδητή Μεταφορά). Στην μη-συνειδητή μεταφορά (oblivious transfer) [5], μια συμβολοσειρά μεταφέρεται μεταξύ ενός αποστολέα και ενός παραλήπτη με τον ακόλουθο τρόπο: ένας αποστολέας έχει δύο συμβολοσειρές S_0 και S_1 . Ο παραλήπτης επιλέγει μια τιμή $i \in \{0, 1\}$ και ο αποστολέας στέλνει την συμβολοσειρά S_i μέσω του πρωτοκόλλου μη-συνειδητής μεταφοράς, έτσι ώστε:

1. Ο παραλήπτης δεν λαμβάνει καμία πληροφορία σχετικά με το $S_{(1-i)}$.
2. Η τιμή του i δεν γίνεται γνωστή στον αποστολέα.

Θα πρέπει να σημειωθεί εδώ ότι ενώ ο παραλήπτης δεν γνωρίζει τις συμβολοσειρές S_0, S_1 , στην πράξη ο παραλήπτης μαθαίνει κάποιες πληροφορίες σχετικά με το τι κωδικοποιεί το κάθε S_i έτσι ώστε ο παραλήπτης να μην επιλέγει τυφλά το i . Δηλαδή, αν το S_0 κωδικοποιεί μια ψευδή τιμή, το S_1 κωδικοποιεί μια αληθή τιμή και ο παραλήπτης θέλει να λάβει την κωδικοποιημένη αληθή τιμή, τότε ο παραλήπτης επιλέγει $i = 1$.

Η μη-συνειδητή μεταφορά μπορεί να υλοποιηθεί χρησιμοποιώντας κρυπτογραφία δημοσίου κλειδιού, όπως το κρυπτοσύστημα RSA [6].

8.1.1.1 Δημιουργία Λογικού Κυκλώματος

Ένα λογικό δυαδικό κύκλωμα, εάν υλοποιεί μια σχετικά μικρή συνάρτηση υπολογισμού μπορεί να δημιουργηθεί με το χέρι, ωστόσο σε αντίθετη περίπτωση έχει ως αποτέλεσμα ένα αρκετά σύνθετο κύκλωμα από XOR και AND πύλες. Σε περιπτώσεις σύνθετων κυκλωμάτων, μπορεί να γίνει χρήση μεθόδων βελτιστοποίησης που δημιουργούν βελτιστοποιημένα κυκλώματα ως προς τον αριθμό των πυλών AND που χρησιμοποιούνται [7]. Όσον αφορά το κύκλωμα που υλοποιεί το πρόβλημα των εκατομμυριούχων, αυτό αποτελεί ουσιαστικά ένα κύκλωμα ψηφιακών συγκριτών, το οποίο αποτελείται από μια αλυσίδα πλήρων αθροιστών (που λειτουργεί ως αφαιρέτης) και εξάγει ένα bit ως έξοδο. Ένα πλήρες κύκλωμα αθροιστή μπορεί να υλοποιηθεί χρησιμοποιώντας μόνο μια πύλη AND και μερικές πύλες XOR. Αυτό σημαίνει ότι ο συνολικός αριθμός των πυλών AND για το κύκλωμα του προβλήματος των εκατομμυριούχων είναι ίσος με το μήκος των bits των τιμών εισόδου.

8.1.1.2 Κρυπτογράφηση Κυκλώματος

Η Αλίκη (ως garbler) κρυπτογραφεί το λογικό κύκλωμα σε αυτό το βήμα για να δημιουργήσει ένα κρυπτογραφημένο κύκλωμα (garbled circuit). Η Αλίκη εκχωρεί δύο συμβολοσειρές, που δημιουργούνται τυχαία και ονομάζονται ετικέτες (labels), σε κάθε καλώδιο του κυκλώματος: μία για την σημασιολογική αναπαράσταση

του 0 και μία για αυτή του 1. Η ετικέτα έχει μήκος k -bits, όπου k είναι μια παράμετρος ασφαλείας που πρέπει να είναι μεγαλύτερη από 80 και συνήθως είναι 128. Στη συνέχεια, πηγαίνει σε όλες τις πύλες του κυκλώματος και αντικαθιστά τα 0 και 1 στους πίνακες αλήθειας της κάθε πύλης με τις αντίστοιχες ετικέτες. Στη συνέχεια κρυπτογραφεί την έξοδο του πίνακα αλήθειας χρησιμοποιώντας τις αντίστοιχες δύο ετικέτες εισόδου. Αυτό γίνεται έτσι ώστε να μπορεί κάποιος να αποκρυπτογραφήσει τον κρυπτογραφημένο πίνακα μόνο εάν γνωρίζει τις σωστές δύο ετικέτες εισόδου. Μετά από αυτό, η Αλίκη ανακατεύει τυχαία τον πίνακα έτσι ώστε η τιμή εξόδου να μην μπορεί να προσδιοριστεί από την γραμμή του πίνακα. Το Σχήμα 8.1 παρουσιάζει της διαδικασία κρυπτογράφησης για την πύλη AND, όπου α και β είναι οι είσοδοι, c η έξοδος, X οι τυχαίες συμβολοσειρές αντικατάστασης, και $Enc_{S_k}(X^c)$ είναι μια συμμετρική κρυπτογράφηση διπλού κλειδιού στην οποία το S_k είναι το μυστικό κλειδί κρυπτογράφησης.

Πίνακας Αλήθειας			Ετικέτες Αντικατάστασης			Κρυπτογραφημένος Πίνακας		
a	b	c	a	b	c			
0	0	0	X_0^a	X_0^b	X_0^c	$Enc_{X_0^a, X_0^b}(X_0^c)$		
0	1	0	X_0^a	X_1^b	X_0^c		$Enc_{X_0^a, X_1^b}(X_0^c)$	
1	0	0	X_1^a	X_0^b	X_0^c		$Enc_{X_1^a, X_0^b}(X_0^c)$	
1	1	1	X_1^a	X_1^b	X_1^c		$Enc_{X_1^a, X_1^b}(X_1^c)$	

Σχήμα 8.1: Παράδειγμα κρυπτογράφησης κυκλώματος για την λογική πύλη AND.

8.1.1.3 Μεταφορά Δεδομένων

Η Αλίκη στέλνει τους κρυπτογραφημένους πίνακες για όλες τις πύλες του κυκλώματος στον Μπάμπη. Ο Μπάμπης χρειάζεται τις ετικέτες εισόδου προκειμένου να αποκρυπτογραφήσει τους πίνακες. Έτσι, η Αλίκη επιλέγει τις ετικέτες που αντιστοιχούν στην είσοδο της a και τις στέλνει στον Μπάμπη. Για παράδειγμα, εάν η είσοδος της Αλίκης είναι $a = a_4a_3a_2a_1a_0 = 01101$, τότε στέλνει τα $X_0^{a_4}, X_1^{a_3}, X_1^{a_2}, X_0^{a_1}$, και $X_1^{a_0}$ στον Μπάμπη. Ο Μπάμπης δεν θα μάθει τίποτα για την είσοδο της Αλίκης, καθώς οι ετικέτες έχουν δημιουργηθεί τυχαία από την Αλίκη και μοιάζουν πραγματικά τυχαίες στον Μπάμπη.

Ο Μπάμπης, στην συνέχεια, χρειάζεται τις ετικέτες που αντιστοιχούν στην δική του είσοδο. Οι ετικέτες αυτές λαμβάνονται από την Αλίκη μέσω ενός πρωτοκόλλου μη-συνειδητής μεταφοράς (Ορισμός 8.2) για κάθε bit της εισόδου του. Για παράδειγμα, εάν η είσοδος του Μπάμπη είναι $b = b_4b_3b_2b_1b_0 = 10100$, ο Μπάμπης πρώτα ζητά το $b_0 = 0$ μεταξύ των ετικετών της Αλίκης $X_0^{b_0}$ και $X_1^{b_0}$. Μέσω της μη-συνειδητής μεταφοράς, λαμβάνει το $X_0^{b_0}$ και ούτω καθεξής. Μετά την εκτέλεση των μη-συνειδητών μεταφορών, η Αλίκη δεν θα μάθει τίποτα για την είσοδο του Μπάμπη και ο Μπάμπης δεν θα μάθει τίποτα για τις άλλες ετικέτες.

8.1.1.4 Αποκρυπτογράφηση Κυκλώματος

Μετά τη μεταφορά δεδομένων, ο Μπάμπης έχει στην διάθεση του τους κρυπτογραφημένους πίνακες και όλες τις ετικέτες εισόδου. Με βάση αυτά, ο Μπάμπης προσπελάσει μία προς μία από όλες τις πύλες του κυκλώματος και προσπαθεί να αποκρυπτογραφήσει τις γραμμές των κρυπτογραφημένων πινάκων. Με αυτόν τον τρόπο, είναι σε θέση να αποκρυπτογραφήσει μια γραμμή για κάθε πίνακα και να ανακτήσει την αντίστοιχη ετικέτα εξόδου: $X^c = Dec_{X^a, X^b}(Κρυπτογραφημένος_Πίνακας[i])$, όπου $0 \leq i \leq 3$. Η διαδικασία αυτή συνεχίζεται για όλους τους πίνακες και μέχρι να φτάσει τελικά στις ετικέτες εξόδου ολόκληρου του κυκλώματος.

8.1.1.5 Αποκάλυψη της Εξόδου

Μετά την αποκρυπτογράφηση ολόκληρου του κυκλώματος των πυλών, ο Μπάμπης αποκτά την ετικέτα εξόδου X^c και η Αλίκη αντίστοιχα γνωρίζει την αντιστοίχιστή της με τις δύο ετικέτες εξόδου: X_0^c και X_1^c . Στην συνέχεια, είτε η Αλίκη μπορεί να μοιραστεί αυτή την πληροφορία με τον Μπάμπη, είτε ο Μπάμπης να αποκαλύψει την έξοδο στην Αλίκη, έτσι ώστε ο ένας ή και οι δύο να μάθουν την τελική έξοδο του κυκλώματος.

8.1.2 Άλλα Πρωτόκολλα MPC

Μετά την εμφάνιση του πρωτοκόλλου GC του Yao [4] για τον ασφαλή υπολογισμό δύο οντοτήτων, άρχισαν να εμφανίζονται αρκετά πρωτόκολλα ασφαλών υπολογισμών πολλαπλών οντοτήτων (MPC), όπως αυτά του Goldreich-Micali-Wigderson (GMW) [8] και του Ben-Or-Goldwasser-Wigderson (BGW) [9]. Ωστόσο, όλα αυτά τα πρωτόκολλα έχουν έναν αριθμό γύρων (round) που εξαρτάται από το βάθος του κυκλώματος υπολογισμού. Σε αντίθεση, το πρωτόκολλο του Beaver-Micali-Rogaway (BMR) [10] εκτελείται με έναν σταθερό αριθμό γύρων, επιτυγχάνοντας παράλληλα ασφάλεια έναντι οποιωνδήποτε t διεφθαρμένων οντοτήτων από τις συνολικά n συμμετέχουσες οντότητες (όπου $t < n$) στο υπολογισμό. Ένα επίσης ενδιαφέρον πρωτόκολλο, που αποτελεί γενίκευση του πρωτοκόλλου του Yao και είναι αρκετά αποτελεσματικό, είναι το GESS (Gate Evaluation Secret Sharing) που προτάθηκε από τον Kolesnikov το 2005 [11]. Στον Πίνακα 8.1 γίνεται μια συγκριτική παρουσίαση των προαναφερθέντων πρωτοκόλλων MPC.

Πίνακας 8.1: Συγκριτική παρουσίαση γνωστών πρωτοκόλλων MPC με παθητικούς αντιπάλους.

Πρωτόκολλο	Οντότητες	Πλήθος Γύρων	Κύκλωμα
GC του Yao [4]	Δύο	Σταθερός αριθμός γύρων	Λογικό δυαδικό κύκλωμα
GMW [8]	Πολλές	Όσο το βάθος του κυκλώματος	Λογικό ή αριθμητικό κύκλωμα
BGW [9]	Πολλές	Όσο το βάθος του κυκλώματος	Λογικό ή αριθμητικό κύκλωμα
BMR [10]	Πολλές	Σταθερός αριθμός γύρων	Λογικό δυαδικό κύκλωμα
GESS [11]	Δύο	Σταθερός αριθμός γύρων	Λογικός τύπος

8.2 Ομομορφική Κρυπτογράφηση

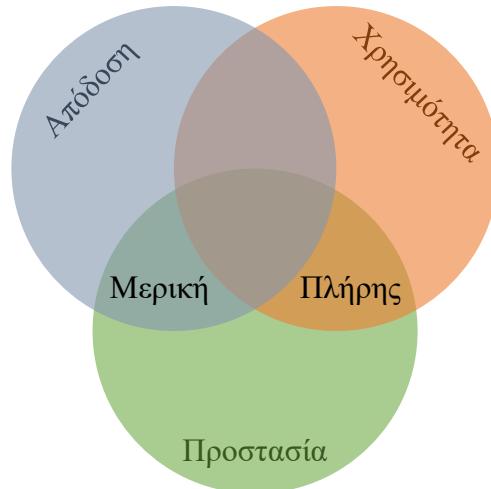
Η ομομορφική κρυπτογράφηση (homomorphic encryption) είναι μια μορφή κρυπτογράφησης που παρέχει τη δυνατότητα πραγματοποίησης υπολογισμών πάνω σε κρυπτογραφημένα δεδομένα χωρίς να υπάρχει πρόσβαση στο ιδιωτικό κλειδί αποκρυπτογράφησης. Το αποτέλεσμα ενός τέτοιου υπολογισμού παραμένει σε κρυπτογραφημένη μορφή. Η ομομορφική κρυπτογράφηση ουσιαστικά αποτελεί μια επέκταση της κρυπτογράφησης δημόσιου (ή ασύμμετρου) κλειδιού. Ο όρος «ομομορφικός» αναφέρεται στον ομομορφισμό της άλγεβρας, όπου οι συναρτήσεις κρυπτογράφησης και αποκρυπτογράφησης μπορούν να θεωρηθούν ως ομομορφισμοί μεταξύ του χώρου του απλού κειμένου (plaintext) και του κρυπτοκειμένου (ciphertext).

Η ομομορφική κρυπτογράφηση περιλαμβάνει πολλούς τύπους κρυπτογράφησης που μπορούν να εκτελέσουν διαφορετικά είδη υπολογισμών σε σχέση με τα κρυπτογραφημένα δεδομένα [12]. Οι υπολογισμοί αυτοί αντιπροσωπεύονται είτε ως λογικές είτε ως αριθμητικές πράξεις. Οι δύο κύριοι τύποι ομομορφικής κρυπτογράφησης, που αναλύονται με περισσότερες λεπτομέρειες στις υποενότητες 8.2.1 και 8.2.2, είναι οι εξής:

- Η μερική ομομορφική κρυπτογράφηση που περιλαμβάνει κρυπτογραφικά σχήματα όπου μπορεί να εκτελεστεί μια μόνο πράξη στα κρυπτογραφημένα δεδομένα, π.χ. πρόσθεση ή πολλαπλασιασμός.
- Η πλήρης ομομορφική κρυπτογράφηση που επιτρέπει ταυτόχρονα πολλαπλές πράξεις (ως επί το πλείστον πρόσθεσης και πολλαπλασιασμού), επιτρέποντας την εκτέλεση περισσότερων και πιο σύνθετων

υπολογισμών σε κρυπτογραφημένα δεδομένα.

Η απόδοση της πλήρους ομομορφικής κρυπτογράφησης είναι επί του παρόντος αρκετά χαμηλή, με αποτέλεσμα απλές πράξεις να χρειάζονται χρόνο από μερικά δευτερόλεπτα έως και ώρες, ανάλογα με τις παραμέτρους ασφαλείας [13]. Επομένως, η ομομορφική κρυπτογράφηση είναι επί του παρόντος μια πράξη εξισορρόπησης μεταξύ χρησιμότητας, προστασίας και απόδοσης. Η πλήρης ομομορφική κρυπτογράφηση παρέχει υψηλή προστασία και χρησιμότητα, αλλά χαμηλή απόδοση. Από την άλλη, η μερική ομομορφική κρυπτογράφηση παρέχει υψηλή απόδοση και προστασία, αλλά πολύ περιορισμένη χρησιμότητα. Αυτό αποτυπώνεται στο Σχήμα 8.2, όπου η τέλεια λύση βρίσκεται στην τομή και των τριών κύκλων.



Σχήμα 8.2: Αποτύπωση της χρησιμότητας, της προστασίας και της απόδοσης στην ομομορφική κρυπτογράφηση.

Στον Πίνακα 8.2 γίνεται μια ιστορική αναδρομή των κυριότερων επιστημονικών δημοσιεύσεων στο χώρο της ομομορφικής κρυπτογράφησης (βάσει του ανάλογου πίνακα από το [14]). Όπως γίνεται αντιληπτό, η έννοια της ομομορφικής κρυπτογράφησης εισήχθη για πρώτη φορά το 1978 με το κρυπτοσύστημα RSA [15]. Το 1985 [16] προτείνεται ένα σχήμα κρυπτογράφησης βάσης δεδομένων που υποστηρίζει τον υπολογισμό ορισμένων στατιστικών πράξεων σε σχέση με κρυπτογραφημένα δεδομένα που είναι αποθηκευμένα στη βάση δεδομένων. Παρόλο που το έγγραφο δεν αναφέρει συγκεκριμένα τον ομομορφισμό, αυτό είναι ένα βήμα για να δείξει ότι η ομομορφική κρυπτογράφηση μπορεί να είναι χρήσιμη στην πράξη. Το 1987 [17] παρέχονται ως παραδείγματα μερικοί αλγόριθμοι που μπορούν να υποστηρίξουν ομομορφικές πράξεις, ωστόσο δεν παρέχουν αρκετή ασφάλεια. Τον επόμενο χρόνο, στο άρθρο [18] αξιολογείται η ασφάλεια της πρώτης προτεινόμενης ομομορφικής κρυπτογράφησης για τον RSA [15] και προτείνεται ένας επιπλέον ομομορφικός αλγόριθμος που μπορεί να παρέχει έναν μέγιστο αριθμό πράξεων πρόσθεσης προτού αυτός αποδιργανωθεί. Το 1996 [19] είδαμε τον πρώτο ομομορφικό αλγόριθμο που υποστηρίζει ταυτόχρονα πράξεις πρόσθεσης και πολλαπλασιασμού, ωστόσο αργότερα το 2003 παρουσιάστηκαν οι αδυναμίες ασφάλειας που είχε [20]. Το πρόβλημα μέχρι αυτό το σημείο ήταν να αναπτυχθεί ένα κρυπτογραφικό σχήμα που είναι ασφαλές, χωρίς να χάσει τις ομομορφικές του ιδιότητες, ικανό να υποστηρίζει την επανάληψη των πράξεων πολλές φορές, και να είναι αποτελεσματικό. Το 2008 [21] προτάθηκε μια λύση που υποστήριζε απεριόριστες πράξεις πρόσθεσης και ταυτόχρονα έναν σταθερό αριθμό πολλαπλασιασμών, ενώ παρέμενε ασφαλές κάτω από ένα γνωστό πρόβλημα αποκωδικοποίησης.

Ωστόσο, η πραγματική επανάσταση συνέβη το 2009 όταν η Gentry [22] πρότεινε ένα κρυπτογραφικό σχήμα που μπορούσε να υποστηρίξει έναν απεριόριστο αριθμό προσθέσεων και πολλαπλασιασμών που η ασφάλειά του βασιζόταν στην ανθεκτικότητα των προβλημάτων πλέγματος (lattice). Έκτοτε, ο Gentry έχει προτείνει και αρκετά άλλα κρυπτογραφικά σχήματα για πλήρη ομομορφική κρυπτογράφηση [23, 24], βελ-

τιώνοντας σταδιακά την λειτουργικότητά τους, αλλά χωρίς να είναι αρκετά αποτελεσματικά για να χρησιμοποιηθούν στον πραγματικό κόσμο. Τέλος, το 2017 προτάθηκε το CKKS [25] που δείχνει να είναι αρκετά αποδοτικό για να εφαρμοστεί στην πράξη σε εφαρμογές μηχανικής μάθησης, ωστόσο το επίπεδο της ανθεκτικότητάς του είναι ακόμη υπό συζήτηση [26].

Πίνακας 8.2: Ιστορική αναδρομή της ομομορφικής κρυπτογράφησης.

Έτος	Τίτλος Επιστημονικής Δημοσίευσης
2017	Homomorphic Encryption for Arithmetic of Approximate Numbers [25]
2013	Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based [24]
2010	Fully Homomorphic Encryption over the Integers [23]
2009	Fully Homomorphic Encryption using Ideal Lattices [22]
2008	A New Approach for Algebraically Homomorphic Encryption [21]
1996	A New Privacy Homomorphism and Applications [19]
1988	On Privacy Homomorphisms [18]
1987	Processing Encrypted Data [17]
1985	A Database Encryption Scheme Which Allows the Computation of Statistics Using Encrypted Data [16]
1978	On Data Banks and Privacy Homomorphisms [15]

8.2.1 Μερική Ομομορφική Κρυπτογράφηση

Η μερική ομομορφική κρυπτογράφηση (Partial Homomorphic Encryption – PHE) εμφανίστηκε για πρώτη φορά το 1978, ακριβώς έναν χρόνο μετά από την εμφάνιση του κρυπτοσυστήματος RSA [15]. Ουσιαστικά αποτελεί μια μορφή κρυπτογράφησης που μπορεί να πραγματοποιήσει μια συγκεκριμένη αλγεβρική πράξη στα μη κρυπτογραφημένα δεδομένα, με την πραγματοποίηση μιας ενδεχομένως διαφορετικής αλγεβρικής πράξης στα κρυπτογραφημένα δεδομένα. Αυτή η ιδιότητα μπορεί να έχει τόσο θετικές όσο και αρνητικές επιπτώσεις σε ένα κρυπτογραφικό σύστημα.

Παρακάτω παρουσιάζονται διάφορα κρυπτογραφικά συστήματα δημοσίου κλειδιού που παρουσιάζουν μερική ομομορφική κρυπτογράφηση μαζί με την αντίστοιχη ομομορφική ιδιότητα που παρουσιάζει είναι:

- **Unpadded RSA** [6]: Εάν το δημόσιο κλειδί είναι (e, n) τότε η κρυπτογράφηση ενός μηνύματος x θα δίνεται από $E(x) = x^e \text{ mod } n$. Οπότε, η ομομορφική ιδιότητα που παρουσιάζει είναι:

$$E(x_1) \cdot E(x_2) = x_1^e x_2^e \text{ mod } n = (x_1 x_2)^e \text{ mod } n = E(x_1 \cdot x_2)$$

- **ElGamal** [27]: Εάν το δημόσιο κλειδί είναι (p, g, y) και το ιδιωτικό κλειδί είναι το a , όπου $y = g^a \text{ mod } p$, τότε η κρυπτογράφηση ενός μηνύματος x θα δίνεται από $E(x) = (g^r, xy^r)$, όπου r ένας τυχαίος αριθμός στο σύνολο $\{1, 2, \dots, p - 1\}$. Οπότε, η ομομορφική ιδιότητα που παρουσιάζει είναι:

$$E(x_1) \cdot E(x_2) = (g^{r_1}, x_1 \cdot y^{r_1}) (g^{r_2}, x_2 \cdot y^{r_2}) = (g^{r_1+r_2}, (x_1 \cdot x_2) y^{r_1+r_2}) = E(x_1 \cdot x_2)$$

- **Goldwasser-Micali** [28]: Εάν το δημόσιο κλειδί είναι (n, x) , όπου n το μόντουλο και x ένα τετραγωνικό μη-υπόλοιπο (quadratic non-residue), τότε η κρυπτογράφηση ενός bit b θα δίνεται από $E(b) = r^2 x^b \text{ mod } n$. Οπότε, η ομομορφική ιδιότητα που παρουσιάζει είναι:

$$E(b_1) \cdot E(b_2) = r_1^2 x^{b_1} r_2^2 x^{b_2} = (r_1 r_2)^2 x^{b_1+b_2} = E(b_1 \oplus b_2)$$

όπου \oplus δηλώνει ένα επιπλέον μόντουλο 2 (π.χ. αποκλειστικό-ή (XOR)).

- **Benaloh** [29]: Εάν το δημόσιο κλειδί είναι (n, g) , όπου n το μόντουλο και g μια βάση με μέγεθος μπλοκ r , τότε η κρυπτογράφηση ενός μηνύματος x θα δίνεται από $E(x) = g^x u^r \text{ mod } n$. Οπότε, η ομομορφική ιδιότητα που παρουσιάζει είναι:

$$E(x_1) \cdot E(x_2) = (g^{x_1} u_1^r) (g^{x_2} u_2^r) = g^{x_1+x_2} (u_1 u_2)^r = E(x_1 + x_2)$$

- **Paillier** [30]: Εάν το δημόσιο κλειδί είναι (n, g) , όπου n το μόντουλο και g μια βάση, τότε η κρυπτογράφηση ενός μηνύματος x θα δίνεται από $E(x) = g^x r^n \text{ mod } n^2$. Οπότε, η ομομορφική ιδιότητα που παρουσιάζει είναι:

$$E(x_1) \cdot E(x_2) = (g^{x_1} r_1^n) (g^{x_2} r_2^n) = g^{x_1+x_2} (r_1 r_2)^n = E(x_1 + x_2)$$

Σε αυτό το σημείο, θα πρέπει ωστόσο να αναφερθεί ότι η πλειοψηφία των εφαρμογών που κάνουν χρήση της μερικής ομομορφικής κρυπτογράφησης βασίζεται στα κρυπτοσυστήματα Paillier [30] και ElGamal [27], αξιοποιώντας την προσθετική και πολλαπλασιαστική ιδιότητα, αντίστοιχα.

8.2.2 Πλήρης Ομομορφική Κρυπτογράφηση

Η πλήρης ομομορφική κρυπτογράφηση (Fully Homomorphic Encryption – FHE), όπως έχει ήδη αναφερθεί, έκανε ουσιαστικά την εμφάνισή της το 2009 με το κρυπτογραφικό σχήμα που προτάθηκε από τον Gentry [22]. Γενικότερα, ένα κρυπτοσύστημα που μπορεί να υποστηρίξει απεριόριστους υπολογισμούς σε κρυπτογραφημένα δεδομένα χαρακτηρίζεται ότι παρέχει πλήρη ομομορφική κρυπτογράφηση. Ένα τέτοιο κρυπτογραφικό σχήμα μπορεί να επιτρέψει την κατασκευή προγραμμάτων για οποιαδήποτε επιθυμητή λειτουργία, η οποία μπορεί να έχει ως είσοδο κρυπτογραφημένα δεδομένα και με στόχο να παράγει ένα επίσης κρυπτογραφημένο αποτέλεσμα. Δεδομένου ότι ένα τέτοιο πρόγραμμα δεν χρειάζεται ποτέ να αποκρυπτογραφήσει τις εισόδους του, μπορεί να εκτελεστεί σε ένα μη έμπιστο περιβάλλον χωρίς να αποκαλύψει τις εισόδους και την εσωτερική του κατάσταση. Τα πλήρη ομομορφικά κρυπτοσυστήματα παρέχουν δυνατότητες για πρακτικές εφαρμογές στην εξωτερική ανάθεση ιδιωτικών υπολογισμών, όπως για παράδειγμα στο υπολογιστικό νέφος [14]. Ακολουθεί μια σύντομη επισκόπηση των πλήρως ομομορφικών σχημάτων κρυπτογράφησης ομαδοποιημένων σε τέσσερις γενιές.

8.2.2.1 Πρώτη Γενιά FHE

Ο Gentry το 2009 [22], χρησιμοποιώντας κρυπτογραφία πλέγματος (lattice) [31], περιέγραψε την πρώτη δυνατή δομή ενός πλήρως ομομορφικού σχήματος κρυπτογράφησης. Το σχήμα αυτό υποστηρίζει ταυτόχρονα πράξεις πρόσθεσης και πολλαπλασιασμού σε κρυπτογραφημένα δεδομένα, παρέχοντας έτσι την δυνατότητα εκτέλεσης απεριόριστων υπολογισμών. Η δομή ξεκινά από ένα «κάπως» (somewhat) ομομορφικό σχήμα κρυπτογράφησης, το οποίο περιορίζεται στον υπολογισμό πολυωνύμων μικρού βαθμού πάνω σε κρυπτογραφημένα δεδομένα. Είναι περιορισμένη επειδή κάθε κρυπτοκείμενο περιέχει θόρυβο με κάποια έννοια, ενώ αυτός ο θόρυβος αυξάνει καθόσον κάποιος προσθέτει και πολλαπλασιάζει κρυπτοκείμενα, έως ότου τελικά ο θόρυβος καταστήσει δυσανάγνωστο το προκύπτον κρυπτοκείμενο.

Στη συνέχεια, ο Gentry παρουσίασε έναν τρόπο ελαφράς τροποποίησης αυτού του σχήματος (με χρήση bootstrapping) έτσι ώστε να το καταστήσει ικανό να αποβάλει τον προστιθέμενο θόρυβο από τα κρυπτοκείμενα. Επιπλέον, έδειξε ότι οποιοδήποτε κρυπτογραφικό σχήμα με bootstrappable «κάπως» ομομορφική κρυπτογράφηση μπορεί να μετατραπεί σε ένα σχήμα πλήρους ομομορφικής κρυπτογράφησης μέσω μιας αναδρομικής αυτο-ενσωμάτωσης. Για το «θορυβώδες» σχήμα του Gentry, η διαδικασία bootstrapping ανανεώνει αποτελεσματικά το κρυπτοκείμενο, εφαρμόζοντας σε αυτό τη διαδικασία αποκρυπτογράφησης ομομορφικά, παρέχοντας έτσι ένα νέο κρυπτογραφημένο κείμενο που κρυπτογραφεί την ίδια τιμή όπως πριν αλλά έχει χαμηλότερο θόρυβο. Με την ανανέωση του κρυπτοκείμενου περιοδικά και όποτε ο θόρυβος μεγαλώνει πολύ, είναι δυνατόν να υπολογιστεί ένας απεριόριστος αριθμός προσθέσεων και πολλαπλασιασμών, χωρίς να αυξηθεί υπερβολικά ο θόρυβος.

Η ασφάλεια αυτού του πρώτου πλήρους ομομορφικού σχήματος κρυπτογράφησης βασίζεται στην υποτιθέμενη δυσκολία δύο προβλημάτων [32]: των προβλημάτων χειρότερης περίπτωσης σε ιδανικά πλέγματα (ideal lattices) και του προβλήματος αθροίσματος του αραιού υποσυνόλου. Αξίζει να σημειωθεί ότι η υλοποίηση του αρχικού κρυπτοσυστήματος του Gentry χρειαζόταν χρονικό διάστημα περίπου 30 λεπτών για μια απλή πράξη ενός bit [13].

Το 2010 παρουσιάστηκε ένα δεύτερο πλήρως ομομορφικό σχήμα κρυπτογράφησης [23], το οποίο χρησιμοποιεί πολλά από τα χαρακτηριστικά του Gentry, αλλά δεν απαιτεί ιδανικά πλέγματα. Σε αυτό παρουσιάζεται ότι η «κάπως» (somewhat) ομομορφική συνιστώσα του ιδανικού πλέγματος του Gentry μπορεί να αντικατασταθεί με ένα πολύ απλό «κάπως» ομομορφικό σχήμα που χρησιμοποιεί ακέραιους αριθμούς. Το σχήμα είναι επομένως εννοιολογικά απλούστερο από το σχήμα ιδανικού πλέγματος, αλλά έχει παρόμοιες ιδιότητες σε σχέση με τις ομομορφικές πράξεις που υποστηρίζει και την αποδοτικότητά του.

8.2.2.2 Δεύτερη Γενιά FHE

Το 2011-12, η δουλειά των Brakerski, Gentry και Vaikuntanathan [33] οδήγησε στην ανάπτυξη πολύ πιο αποτελεσματικών «κάπως» και πλήρης ομομορφικών κρυπτοσυστημάτων:

- Το κρυπτοσύστημα BGV [33].
- Το κρυπτοσύστημα LTV [34] που βασίζεται στο κρυπτοσύστημα NTRU [35].
- Το κρυπτοσύστημα BFV [36] που βασίζεται στο κρυπτοσύστημα Brakerski [37].
- Το κρυπτοσύστημα BLLN [38] που βασίζεται στο κρυπτοσύστημα LTV [34] και στο Brakerski [37].

Η ασφάλεια των περισσότερων από αυτά τα σχήματα βασίζεται στη δυσκολία του προβλήματος RLWE (Ring Learning With Errors), εκτός από τα σχήματα LTV και BLLN που βασίζονται σε μια παραλλαγή [39] του υπολογιστικού προβλήματος NTRU. Ωστόσο, αυτή η παραλλαγή του NTRU φαίνεται να είναι ευάλωτη σε επιθέσεις [40] και για αυτό τον λόγο αυτά τα δύο σχήματα δεν χρησιμοποιούνται πλέον στην πράξη.

Όλα τα κρυπτοσυστήματα της δεύτερης γενιάς εξακολουθούν να ακολουθούν το βασικό προσχέδιο της αρχικής δομής του Gentry, δηλαδή πρώτα δημιουργούν ένα «κάπως» ομομορφικό κρυπτοσύστημα και μετά το μετατρέπουν σε ένα πλήρως ομομορφικό κρυπτοσύστημα χρησιμοποιώντας bootstrapping. Ένα επιπλέον σημαντικό χαρακτηριστικό των κρυπτοσυστημάτων δεύτερης γενιάς είναι ότι όλα παρουσιάζουν πολύ πιο αργή ανάπτυξη θορύβου κατά τη διάρκεια των ομομορφικών υπολογισμών.

8.2.2.3 Τρίτη Γενιά FHE

Το 2013, οι Gentry, Sahai και Waters (GSW) πρότειναν το κρυπτοσύστημα GSW [24] που ενσωματώνει μια νέα τεχνική για την κατασκευή πλήρως ομομορφικών κρυπτοσυστημάτων αποφεύγοντας το υπολογιστικά ακριβό βήμα «επαγραμμικοποίησης» (relinearization) στον ομομορφικό πολλαπλασιασμό. Επιπλέον, παρατηρήθηκε ότι για ορισμένους τύπους πράξεων, το κρυπτοσύστημα GSW παρουσιάζει έναν ακόμη πιο αργό ρυθμό αύξησης του θορύβου, και επομένως παρέχει καλύτερη απόδοση και ισχυρότερη προστασία [41]. Στην συνέχεια προτάθηκε μια ακόμη πιο αποτελεσματική τεχνική bootstrapping με βάση αυτήν την παρατήρηση [42].

Αυτές οι τεχνικές βελτιώθηκαν περαιτέρω για την ανάπτυξη αποτελεσματικών παραλλαγών δακτυλίου του κρυπτοσυστήματος GSW: το FHEW [43] και το TFHE [44]. Το κρυπτογραφικό σχήμα FHEW ήταν το πρώτο που έδειξε ότι με την ανανέωση των κρυπτοκειμένων μετά από κάθε πράξη, είναι δυνατό να μειωθεί ο χρόνος του bootstrapping σε ένα κλάσμα του δευτερολέπτου. Επίσης, το FHEW πρότεινε μια νέα μέθοδο για τον υπολογισμό λογικών πράξεων σε κρυπτογραφημένα που απλοποιεί σε μεγάλο βαθμό το bootstrapping εφαρμόζοντας μια παραλλαγή της διαδικασίας αυτής [42]. Η αποδοτικότητα του FHEW βελτιώ-

θηκε περαιτέρω με το κρυπτοσύστημα TFHE, το οποίο εφαρμόζει μια παραλλαγή δακτυλίου της διαδικασίας bootstrapping [45] χρησιμοποιώντας μια μέθοδο παρόμοια με αυτή του FHEW.

8.2.2.4 Τέταρτη Γενιά FHE

Το κρυπτοσύστημα CKKS [25] αποτελεί ότι πιο σύγχρονο υπάρχει στην πλήρη ομομορφική κρυπτογράφηση και υποστηρίζει αποτελεσματικά πράξεις στρογγυλοποίησης σε κρυπτογραφημένη κατάσταση. Η πράξη στρογγυλοποίησης ελέγχει την αύξηση του θορύβου στον κρυπτογραφημένο πολλαπλασιασμό, γεγονός που μειώνει τον αριθμό των bootstrappings που απαιτούνται. Αυτό οφείλεται σε ένα χαρακτηριστικό του σχήματος CKKS που κρυπτογραφεί τις κατά προσέγγιση και όχι τις ακριβείς τιμές. Όταν οι υπολογιστές αποθηκεύουν δεκαδικές τιμές, θυμούνται κατά προσέγγιση τιμές με τα περισσότερο σημαντικά bits και όχι τιμές με άπειρο πλήθος δεκαδικών ψηφίων. Το σχήμα CKKS έχει σχεδιαστεί για να αντιμετωπίζει αποτελεσματικά τα σφάλματα που προκύπτουν από τέτοιου είδους στρογγυλοποίησεις και μπορεί να εφαρμοστεί στη μηχανική μάθηση που διαθέτει έτσι και αλλιώς τέτοιους θορύβους.

Ωστόσο, ένα άρθρο του 2021 [46] συζητά παθητικές επιθέσεις εναντίον του CKKS, υποδηλώνοντας ότι ο τυπικός ορισμός της επίθεσης IND-CPA ενδέχεται να μην είναι επαρκής σε σενάρια όπου μοιράζονται τα αποτελέσματα αποκρυπτογράφησης. Πιο αναλυτικά, η επίθεση αυτή είχε ως αποτέλεσμα σε τέσσερις σύγχρονες βιβλιοθήκες ομομορφικής κρυπτογράφησης (HEAAN, SEAL, HElib και PALISADE) να ανακτηθεί το μυστικό κλειδί από τα αποτελέσματα αποκρυπτογράφησης με διάφορες παραμετροποίησεις των βιβλιοθηκών αυτών. Τέλος, αξίζει να αναφερθεί ότι έχουν ήδη προταθεί και εφαρμοστεί διάφορες στρατηγικές μετριασμού αυτού του τύπου επιθέσεων στις αναφερόμενες βιβλιοθήκες [26].

8.3 Αποδείξεις Μηδενικής Γνώσης

Οι αποδείξεις μηδενικής γνώσης (Zero Knowledge Proofs – ZKP) προτάθηκαν στη δεκαετία του 1980 από τους Shafi Goldwasser, Silvio Micali και Charles Rackoff [47] και αποτέλεσαν μια έννοια η οποία έδωσε πολλές εφαρμογές, τόσο θεωρητικές όσο και πρακτικές, λόγος για τον οποίο οι δύο πρώτοι συγγραφείς βραβεύθηκαν με το βραβείο Turing το 2013. Προτάθηκαν ως μια παραλλαγή των διαλογικών συστημάτων αποδείξεων (interactive proof systems) [48], στα οποία ένας υπολογισμός υλοποιείται με ανταλλαγή μηνυμάτων μεταξύ μιας οντότητας η οποία ονομάζεται αποδεικνύων (Prover – P) και μιας οντότητας η οποία ονομάζεται επιβεβαιωτής (Verifier – V). Τυπικά, ο P θέλει να πείσει τον V ότι μία συμβολοσειρά μάρτυρας (witness) ανήκει σε μια γλώσσα, ή ισοδύναμα ότι μια πρόταση είναι αληθής. Ο P και ο V αναπαριστώνται ως πιθανοτικές μηχανές Turing. Συνήθως ο P έχει απεριόριστη υπολογιστική ισχύ, ενώ ο V περιορίζεται σε πιθανοτικούς υπολογισμούς πολυωνυμικής πολυπλοκότητας χρόνου (Probabilistic Polynomial Time – PPT).

Σε ένα οποιοδήποτε σύστημα αποδείξεων είναι επιθυμητές οι εξής δύο ιδιότητες:

- Πληρότητα:** Όλες οι αληθείς προτάσεις μπορούν να αποδειχθούν. Σε ένα αλληλεπιδραστικό σύστημα επομένως, ένας τίμιος P (που όντως κατέχει μια συμβολοσειρά που ανήκει σε μια γλώσσα) πείθει ένα τίμιο V (που δηλαδή ακολουθεί ακριβώς το πρωτόκολλο) με πολύ μεγάλη πιθανότητα.
- Ορθότητα:** Οι ψευδείς προτάσεις δεν μπορούν να αποδειχθούν. Δηλαδή, ένας κακόβουλος P, που προσπαθεί να αποδείξει μια ψευδή πρόταση, δεν μπορεί να πείσει ένα τίμιο V, παρά μόνο με αμελητέα πιθανότητα.

Οι Goldwasser, Micali και Rackoff [47] ασχολήθηκαν με το πόση πληροφορία διαρρέει ένα τέτοιο σύστημα. Με άλλα λόγια, διερεύνησαν τι επιπλέον μαθαίνει ο V πέρα από το γεγονός ότι ο ισχυρισμός του P είναι έγκυρος. Έτσι προσέθεσαν μία τρίτη ιδιότητα στα διαλογικά συστήματα:

- Μηδενική Γνώση:** Ο V δεν μαθαίνει τίποτε παραπάνω από το γεγονός ότι ο ισχυρισμός του P είναι αληθής.

Κομβικό ρόλο στην απόδειξη του ότι ένα διαλογικό σύστημα έχει την ιδιότητα της μηδενικής γνώσης διαδραματίζει ο προσομοιωτής (*Simulator – S*), ο οποίος προσομοιώνει τον *P*, χωρίς όμως να έχει πρόσβαση στην συμβολοσειρά μάρτυρα (*witness*). Η συνεισφορά του είναι η εξής: Ο *V* αλληλεπιδρά με τον *S*. Κάποια στιγμή ο *V* θα φέρει τον *S* στην «δύσκολη θέση» να μην μπορεί να απαντήσει ένα ερώτημα, καθώς δεν έχει πρόσβαση στον μάρτυρα. Σε αυτή την περίπτωση επαναφέρουμε τον *V* σε μια κατάσταση πριν την δυσάρεστη ερώτηση και τρέχουμε το πρωτόκολλο από εκείνο το σημείο και μετά. Αν τελικά ο *V* (με συνεχείς επαναφορές κατάστασης) αποδεχθεί την απόδειξη του *S*, το πρωτόκολλο κατέχει την ιδιότητα της μηδενικής γνώσης, καθώς ο *V* δεν μπορεί να ξεχωρίσει έναν *P* που γνωρίζει τον μάρτυρα και έναν *S* που υποκρίνεται. Ο *V* δηλαδή δεν εξάγει καμία επιπλέον πληροφορία από το πρωτόκολλο (αφού στην δεύτερη περίπτωση δεν υπάρχει πληροφορία για να εξαχθεί).

Παραδείγματα τέτοιων διαλογικών αποδείξεων μηδενικής γνώσης, μεταξύ αρκετών άλλων [49], είναι τα εξής δύο πρωτόκολλα:

- Το πρωτόκολλο του Schnorr (Ενότητα 8.3.1): Απόδειξη γνώσης του διακριτού λογάριθμου.
- Το πρωτόκολλο Chaum-Pedersen (Ενότητα 8.3.2): Απόδειξη ισότητας δύο διακριτών λογάριθμων.

Εκτός από την ύπαρξη διαλογικών πρωτοκόλλων αποδείξεων μηδενικής γνώσης, υπάρχουν και μη-διαλογικά πρωτόκολλα αποδείξεων μηδενικής γνώσης [50] τα οποία δεν απαιτούν την ύπαρξη κάποιου επιβεβαιωτή *V*. Αυτό είναι δυνατόν να επιτευχθεί με την αντικατάσταση της τυχαίας πρόκλησης (*challenge*) του *V* κάνοντας χρήση του αποτελέσματος μιας ψευδοτυχαίας συνάρτησης με είσοδο τη δέσμευση (*commitment*) του *P*, όπως για παράδειγμα με τη χρήση μιας συνάρτησης σύνοψης.

8.3.1 Το Πρωτόκολλο του Schnorr

Μια από τις πιο απλές και συχνά χρησιμοποιούμενες αποδείξεις μηδενικής γνώσης είναι η απόδειξη της γνώσης ενός διακριτού λογάριθμου κάνοντας χρήση του πρωτοκόλλου του Schnorr [51]. Το πρωτόκολλο αυτό ορίζεται για μια κυκλική ομάδα G_q , τάξης q και με γεννήτορα g .

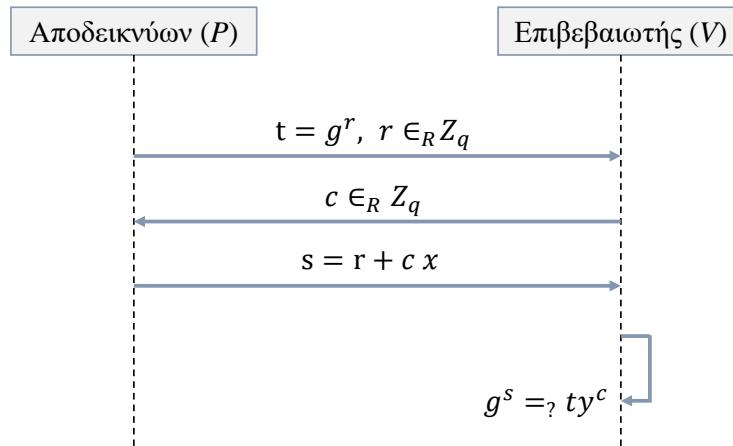
Προκειμένου να αποδειχθεί η γνώση του διακριτού λογάριθμου $x = \log_g y$ (ή $y = g^x$), ο αποδεικνύων *P* αλληλεπιδρά με τον επιβεβαιωτή *V* ως εξής (Σχήμα 8.3):

1. Στον πρώτο βήμα ο *P* δεσμεύεται σε μια τυχαία τιμή r . Επομένως, αποστέλλει στον *V* το μήνυμα $t = g^r$ το οποίο καλείται δέσμευση (*commitment*).
2. Ο *V* απαντά με μια πρόκληση (*challenge*) c που επιλέχθηκε τυχαία.
3. Ο *P* μετά την λήψη του c στέλνει ως απάντηση (*response*) το τρίτο και τελευταίο μήνυμα $s = r + cx$ με μόντουλο της τάξης της ομάδας.
4. Ο *V* δέχεται την απάντηση μόνο εάν $g^s = ty^c$.

Μπορούμε να δούμε ότι αυτό αποτελεί μια έγκυρη απόδειξη μηδενικής γνώσης επειδή εάν είχαμε έναν προσομοιωτή *S* θα λειτουργούσε ως εξής:

1. Προσομοιώνει τον *P* για την έξοδο $t = g^r$. Ο *P* βρίσκεται τώρα στην κατάσταση *Q*.
2. Δημιουργεί μια τυχαία τιμή c_1 και την εισάγει στον *P*. Η έξοδος του *P* θα είναι $s_1 = r + c_1x$.
3. Επαναφέρει τον *P* στην κατάσταση *Q*. Τώρα δημιουργεί μια διαφορετική τυχαία τιμή c_2 και την εισάγει στον *P* για να λάβει το $s_2 = r + c_2x$.
4. Η έξοδος υπολογίζεται ως εξής $(s_1 - s_2)(c_1 - c_2)^{-1}$.

Από την στιγμή που $(s_1 - s_2) = (r + c_1x) - (r + c_2x) = x(c_1 - c_2)$, η έξοδος του προσομοιωτή είναι το ίδιο το x .



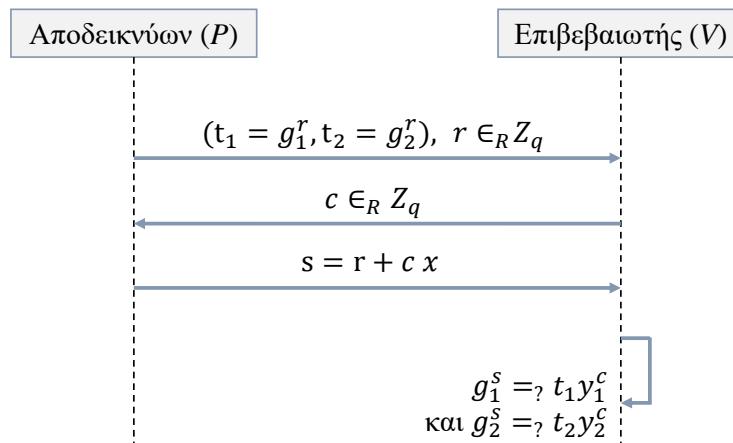
Σχήμα 8.3: Τα βήματα εκτέλεσης του πρωτοκόλλου του Schnorr.

8.3.2 Το Πρωτόκολλο Chaum-Pedersen

Μια παραλλαγή του πρωτοκόλλου του Schnorr είναι το πρωτόκολλο Chaum-Pedersen [52] με το οποίο μπορεί να αποδειχτεί η ισότητα δύο διακριτών λογάριθμων. Οι είσοδοι του πρωτοκόλλου είναι οι γεννήτορες g_1 και g_2 μιας τάξης q .

Προκειμένου ο αποδεικνύων P να αποδείξει ότι γνωρίζει $x \in Z_q$ τέτοιο ώστε $y_1 = g_1^x$ και $y_2 = g_2^x$ (δηλ., $x = \log_{g_1} y_1 = \log_{g_2} y_2$), χωρίς φυσικά να αποκαλύψει το x , αλληλεπιδρά με τον επιβεβαιωτή V ως εξής (Σχήμα 8.4):

- Στον πρώτο βήμα ο P δεσμεύεται σε μια τυχαία τιμή r . Επομένως, αποστέλλει στον V τα μηνύματα $t_1 = g_1^r$ και $t_2 = g_2^r$.
- Ο V απαντά με μια πρόκληση (challenge) c που επιλέχθηκε τυχαία.
- Ο P μετά την λήψη του c στέλνει ως απάντηση (response) το τρίτο και τελευταίο μήνυμα $s = r + cx$ με μόντουλο της τάξης της ομάδας.
- Ο V δέχεται την απάντηση μόνο εάν $g_1^s = t_1 y_1^c$ και $g_2^s = t_2 y_2^c$.



Σχήμα 8.4: Τα βήματα εκτέλεσης του πρωτοκόλλου του Chaum-Pedersen.

Το πρωτόκολλο Chaum-Pedersen μπορεί επίσης να χρησιμοποιηθεί για να αποδείξει ότι ένα κρυπτοκείμενο ElGamal $(G', M') = (Gy^s, Mg^s)$ είναι μια επανακρυπτογράφηση του κρυπτοκείμενου $(G, M) = (my^r, g^r)$ χωρίς να αποκαλύπτεται ο συντελεστής τυχαιοποίησης s , γεγονός που υποδηλώνει ότι $\log_y(G'/G) = \log_g(M'/M)$. Επιπλέον, το πρωτόκολλο Chaum-Pedersen μπορεί να χρησιμοποιηθεί για να αποδείξει ότι ένα κρυπτοκείμενο ElGamal έχει αποκρυπτογραφηθεί σωστά.

8.4 Ψηφιακές Υπογραφές Ενίσχυσης του Απορρήτου

Οι ψηφιακές υπογραφές έχουν ως στόχο να εξασφαλίσουν την αυθεντικότητα και την ακεραιότητα των ηλεκτρονικών επικοινωνιών και να αποφευχθεί η αποποίηση πραγματοποίησης μιας πράξης. Οι ψηφιακές υπογραφές έχουν καταστεί δυνατές με την ανάπτυξη των κρυπτογραφικών συστημάτων δημόσιου κλειδιού. Επιπρόσθετα, οι ψηφιακές υπογραφές και σε συνδυασμό με τις υποδομές δημοσίου κλειδιού μπορούν να χρησιμοποιηθούν για την παροχή ελέγχου ταυτότητας των ατόμων. Στην ενότητα αυτή, ωστόσο, απαριθμούμε συγκεκριμένες κατηγορίες ψηφιακών υπογραφών που επιτρέπουν τόσο τον έλεγχο ταυτότητας, όσο και την ενίσχυση της ιδιωτικότητας/απορρήτου του χρήστη.

8.4.1 Τυφλές Υπογραφές

Μια τυφλή υπογραφή (blind signature), όπως αρχικά ορίστηκε από τον David Chaum [53], αποτελεί μια μορφή ψηφιακής υπογραφής στην οποία το περιεχόμενο ενός μηνύματος συγκαλύπτεται (blinded) πριν υπογραφεί. Η τυφλή υπογραφή που προκύπτει μπορεί να επαληθευτεί δημοσίως έναντι του αρχικού, μη τυφλού (unblinded) μηνύματος, όπως ακριβώς γίνεται σε μια κανονική ψηφιακή υπογραφή. Οι τυφλές υπογραφές χρησιμοποιούνται συνήθως σε πρωτόκολλα που σχετίζονται με το απόρρητο, όπου ο υπογράφων (signer) και ο συντάκτης του μηνύματος είναι διαφορετικές οντότητες. Παραδείγματα εφαρμογών των τυφλών υπογραφών αποτελούν τα σχήματα εκλογικών συστημάτων και των ψηφιακών χρημάτων.

Οι τυφλές υπογραφές μπορούν επίσης να χρησιμοποιηθούν για την παροχή μη-συνδεσιμότητας (unlinkability), γεγονός που εμποδίζει τον υπογράφοντα να συνδέσει το συγκαλυμμένο (blinded) μήνυμα που υπογράφει σε μια μεταγενέστερη «μη τυφλή» έκδοση που μπορεί να κληθεί να επαληθεύσει. Σε αυτήν την περίπτωση, στην απάντηση του υπογράφοντα έχει αφαιρεθεί η συγκάλυψη του μηνύματος πριν από την επαλήθευση, με τέτοιο τρόπο, ώστε η υπογραφή να παραμένει έγκυρη για το αρχικό μήνυμα. Μια τέτοια ιδιότητα μπορεί να είναι εξαιρετικά χρήσιμη σε σχήματα όπου απαιτείται ανωνυμία.

Τα σχήματα τυφλών υπογραφών μπορούν να υλοποιηθούν κάνοντας χρήση διαφόρων γνωστών κρυπτοσυστημάτων υπογραφής δημοσίου κλειδιού, όπως για παράδειγμα του RSA [6] και του DSA [54]. Για την πραγματοποίηση μιας τέτοιας υπογραφής, το μήνυμα αρχικά συγκαλύπτεται (blinded), συνήθως συνδυάζοντάς το με κάποιον τρόπο με έναν τυχαίο «παράγοντα συγκάλυψης» (blinding factor). Στην συνέχεια, το συγκαλυμμένο μήνυμα διαβιβάζεται σε έναν υπογράφοντα, ο οποίος στη συνέχεια το υπογράφει χρησιμοποιώντας έναν συνηθισμένο αλγόριθμο υπογραφής. Το μήνυμα που προκύπτει, μαζί με τον παράγοντα συγκάλυψης, μπορεί αργότερα να επαληθευτεί με το δημόσιο κλειδί του υπογράφοντος. Σε ορισμένα σχήματα τυφλής υπογραφής, όπως του RSA, είναι ακόμη δυνατό να αφαιρεθεί ο παράγοντας συγκάλυψης από την υπογραφή πριν από την επαλήθευσή της. Σε αυτά τα σχήματα, η τελική έξοδος (μήνυμα/υπογραφή) του σχήματος τυφλής υπογραφής είναι πανομοιότυπη με εκείνη του κανονικού αλγορίθμου υπογραφής.

Ως παράδειγμα, παρουσιάζεται το πιο απλό σχήμα τυφλών υπογραφών το οποίο βασίζεται στην κρυπτογράφηση RSA. Ο υπογράφων έχει στην κατοχή του ένα δημόσιο κλειδί (e, n) και ένα ιδιωτικό κλειδί d . Ας υποθέσουμε ότι μια οντότητα A θέλει να έχει ένα μήνυμα m υπογεγραμμένο χρησιμοποιώντας μια τυφλή υπογραφή, ωστόσο δεν θέλει να κάνει γνωστό το μήνυμα m στον υπογράφοντα. Τα βήματα του πρωτοκόλλου που θα πρέπει να εκτελεστούν με τον υπογράφοντα S είναι τα εξής:

1. Η οντότητα A πρώτα επιλέγει τυχαία μια τιμή r , η οποία ικανοποιεί τις εξής ιδιότητες: $0 \leq r \leq n - 1$ και $\gcd(n, r) = 1$.

2. Για το μήνυμα m , η οντότητα A υπολογίζει το $m' = mr^e \pmod{n}$ και στέλνει το m' στον υπογράφοντα S .
3. Όταν ληφθεί το μήνυμα m' από τον υπογράφοντα S , υπολογίζει το $s' = (m')^d \pmod{n}$ και στέλνει το s' πίσω στην οντότητα A .
4. Η οντότητα A υπολογίζει το $s = s' \cdot r^{-1} \pmod{n}$. Η υπογραφή s ουσιαστικά αποτελεί την υπογραφή του S για το μήνυμα m , γιατί $r^{ed} \equiv r \pmod{n}$.

Ωστόσο, στο απλό αυτό σχήμα που περιγράφηκε παραπάνω, το αρχικό μήνυμα (m) και η μη τυφλή υπογραφή (s) είναι έγκυρη, αλλά το ίδιο ισχύει και για το συγκαλυμμένο μήνυμα (m') και τη τυφλή υπογραφή (s'), και πιθανώς άλλους συνδυασμούς τους οποίους μπορεί να παράγει ένας έξυπνος επιτιθέμενος. Στην πράξη, μια λύση σε αυτό το πρόβλημα είναι να υπογραφεί τυφλά η σύνοψη (hash) του μηνύματος, όχι το ίδιο το μήνυμα [55].

8.4.2 Ομαδικές Υπογραφές

Μια ομαδική υπογραφή (group signature) αποτελεί ένα κρυπτογραφικό σχήμα υπογραφών που επιτρέπει σε ένα μέλος μιας ομάδας να υπογράφει ανώνυμα ένα μήνυμα για λογαριασμό της ομάδας. Η ιδέα εισήχθη για πρώτη φορά από τους David Chaum και Eugene van Heyst το 1991 [56]. Για παράδειγμα, ένα σχήμα ομαδικής υπογραφής θα μπορούσε να χρησιμοποιηθεί από έναν υπάλληλο μιας μεγάλης εταιρείας, όπου είναι αρκετό για έναν επαληθευτή (verifier) να γνωρίζει ότι ένα μήνυμα έχει υπογραφεί από κάποιον υπάλληλο της εταιρίας, αλλά δεν χρειάζεται να γνωρίζει από ποιον συγκεκριμένο υπάλληλο. Μια άλλη εφαρμογή των ομαδικών υπογραφών είναι για την παροχή πρόσβασης σε περιορισμένες περιοχές όπου δεν είναι θεμιτό να παρακολουθούνται οι ακριβείς κινήσεις μεμονωμένων εργαζομένων, αλλά είναι απαραίτητο να διασφαλίζεται η πρόσβαση μόνο σε εργαζομένους της ομάδας.

Εξαιρετικά σημαντική για ένα σχήμα ομαδικών υπογραφών είναι η ύπαρξη ενός διαχειριστή ομάδας, ο οποίος είναι υπεύθυνος για την προσθήκη μελών στην ομάδα και έχει τη δυνατότητα να αποκαλύψει τον αρχικό υπογράφοντα σε μια περίπτωση διαφωνίας. Σε ορισμένα συστήματα η ευθύνη προσθήκης μελών και ανάκλησης της ανωνυμίας μιας υπογραφής διαχωρίζονται και ανατίθενται σε δύο διαφορετικές οντότητες, τον διαχειριστή μελών και τον διαχειριστή ανακλήσεων, αντίστοιχα [57]. Μετά την πρώτη εμφάνιση των ομαδικών υπογραφών έχουν προταθεί αρκετά νέα σχήματα, όπως του ACJT00 [58] και του BBS04 [59], ωστόσο όλα αυτά θα πρέπει να υποστηρίζουν τις ακόλουθες βασικές απαιτήσεις:

- **Ορθότητα και πληρότητα:** Οι έγκυρες υπογραφές από τα μέλη της ομάδας επαληθεύονται πάντα σωστά και οι μη έγκυρες υπογραφές αποτυγχάνουν πάντα στην επαλήθευση.
- **Αδυναμία πλαστογράφησης (unforgeable):** Μόνο τα μέλη της ομάδας μπορούν να δημιουργήσουν έγκυρες υπογραφές ομάδας.
- **Ανωνυμία:** Δεδομένου ενός μηνύματος και της υπογραφής του, η ταυτότητα ενός μεμονωμένου υπογράφοντος δεν μπορεί να προσδιοριστεί χωρίς το μυστικό κλειδί του διαχειριστή ομάδας.
- **Ιχνηλασμότητα:** Με δεδομένη οποιαδήποτε έγκυρη υπογραφή, ο διαχειριστής ομάδας θα πρέπει να μπορεί να εντοπίσει ποιος χρήστης εξέδωσε την υπογραφή.
- **Αδυναμία συσχέτισης (unlinkability):** Δεδομένου δύο μηνυμάτων και των υπογραφών τους, δεν μπορούμε να πούμε εάν οι υπογραφές προέρχονται από τον ίδιο υπογράφοντα ή όχι.
- **Αδυναμία πλαστογράφησης υπογραφής για μη συμμετέχοντα:** Ακόμα κι αν όλα τα μέλη της ομάδας (και οι διαχειριστές) συνωμοτήσουν, δεν θα μπορέσουν να πλαστογράφησουν μια υπογραφή για ένα μη συμμετέχον μέλος της ομάδας.

- **Αδυναμία πλαστογράφησης της επαλήθευσης εντοπισμού:** Ο διαχειριστής ανακλήσεων δεν μπορεί να κατηγορήσει ψευδώς έναν υπογράφοντα ότι δημιούργησε μια υπογραφή χωρίς να το έχει κάνει ο ίδιος.
- **Ανθεκτικότητα σε συνασπισμούς (coalitions):** Ένα αθέμιτος συνασπισμός ενός υποσυνόλου των μελών της ομάδας δεν μπορεί να δημιουργήσει μια έγκυρη υπογραφή που ο διαχειριστής ομάδας δεν μπορεί να συσχετίσει με ένα από τα μέλη της ομάδας.

8.4.3 Υπογραφές Δακτυλίου

Η υπογραφή δακτυλίου (ring signature) αποτελεί έναν τύπο ψηφιακής υπογραφής που μπορεί να πραγματοποιηθεί από οποιοδήποτε μέλος ενός συνόλου χρηστών που ο καθένας διαθέτει κλειδιά. Επομένως, ένα μήνυμα που υπογράφεται με υπογραφή δακτυλίου επικυρώνεται από κάποιον σε ένα συγκεκριμένο σύνολο ατόμων. Μια από τις ιδιότητες ασφαλείας μιας υπογραφής δαχτυλιδιού είναι ότι θα πρέπει να είναι υπολογιστικά αδύνατο να προσδιοριστεί ποιο από τα κλειδιά του συνόλου των μελών χρησιμοποιήθηκε για την δημιουργία της υπογραφής. Οι υπογραφές δακτυλίου είναι παρόμοιες με τις ομαδικές υπογραφές (Ενότητα 8.4.2), αλλά διαφέρουν σε δύο βασικά σημεία: (1) δεν υπάρχει τρόπος να ανακληθεί η ανωνυμία μιας μεμονωμένης υπογραφής, και (2) οποιοδήποτε σύνολο χρηστών μπορεί να χρησιμοποιηθεί ως σύνολο υπογραφής χωρίς κάποια επιπρόσθετη ρύθμιση.

Οι υπογραφές δακτυλίου προτάθηκαν για πρώτη φορά από τους Rivest, Shamir και Tauman το 2001 [60], και το όνομα τους προήλθε από την δακτυλιοειδή δομή που παρουσιάζει ο αλγόριθμος υπογραφής δακτυλίου (βλέπε Σχήμα 8.5). Οι πρότυπες υπογραφές δακτυλίου που προτάθηκαν βασίζονται σε υπογραφές RSA [61], αλλά και σε υπογραφές Rabin [62]. Οι οποίες ορίζονται μια συνάρτηση συνδυασμού κλειδιών $C_{k,v}(y_1, y_2, \dots, y_n)$ η οποία απαιτεί ένα κλειδί k , μια τιμή αρχικοποίησης v , και μια λίστα με τιμές y_1, \dots, y_n . Μια τιμή y_i ορίζεται ως $g_i(x_i)$, όπου g_i είναι μια συνάρτηση καταπακτής (trap-door) (όπως π.χ. το δημόσιο κλειδί του RSA).

Η συνάρτηση $C_{k,v}(y_1, y_2, \dots, y_n)$, που ονομάζεται εξίσωση δακτυλίου, βασίζεται σε μια συμμετρική συνάρτηση κρυπτογράφησης E_k , και ορίζεται ως εξής:

$$C_{k,v}(y_1, y_2, \dots, y_n) = E_k(y_n \oplus E_k(y_{n-1} \oplus E_k(\dots \oplus E_k(y_1 \oplus v) \dots)))$$

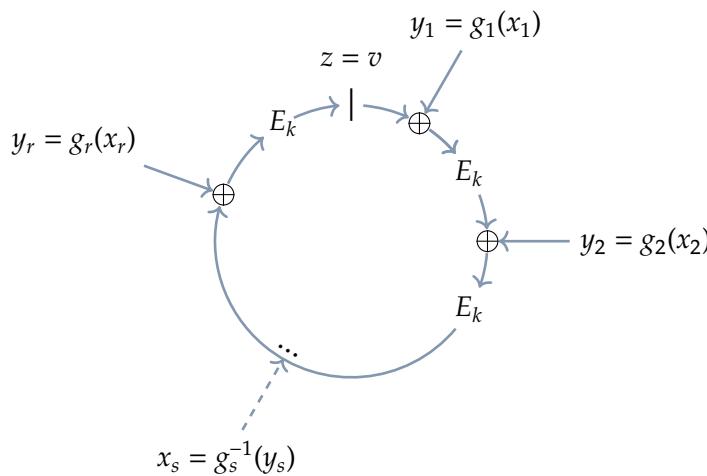
Η έξοδος αυτής της συνάρτησης είναι μια τιμή z η οποία υπολογίζεται έτσι ώστε να είναι ίση με το v . Η εξίσωση $v = C_{k,v}(y_1, y_2, \dots, y_n)$ μπορεί να λυθεί εφόσον υπάρχει τουλάχιστον ένα y_i , και κατ' επέκταση ένα x_i , που μπορεί να επιλεγεί ελεύθερα. Σύμφωνα με τις παραδοχές του RSA, αυτό συνεπάγεται γνώση τουλάχιστον μίας από τις αντίστροφες συναρτήσεις καταπακτής g_i^{-1} (δηλ., το ιδιωτικό κλειδί του RSA), δεδομένου ότι $g_i^{-1}(y_i) = x_i$.

Δημιουργία Υπογραφών

Η δημιουργία μιας υπογραφής δακτυλίου (Σχήμα 8.5) για ένα μήνυμα m και με δημόσια κλειδιά δακτυλίου τα P_1, P_2, \dots, P_n , πραγματοποιείται με τα ακόλουθα έξι βήματα:

1. Υπολογισμός του κλειδιού $k = H(m)$, κάνοντας χρήση μιας συνάρτησης σύνοψης. Αυτό το βήμα προϋποθέτει μια αρκετά ασφαλή συνάρτηση σύνοψης H , αφού το k θα χρησιμοποιηθεί ως κλειδί για την συμμετρική κρυπτογράφηση E_k .
2. Επιλογή μιας τυχαίας τιμής αρχικοποίησης v .
3. Επιλογή τυχαίων x_i για όλα τα μέλη του δακτυλίου εκτός από τον υπογράφοντα (το x_s θα υπολογιστεί χρησιμοποιώντας το ιδιωτικό κλειδί του υπογράφοντος), και υπολογισμός των αντίστοιχων $y_i = g_i(x_i)$.
4. Λύση της εξίσωσης δακτυλίου για y_s .
5. Υπολογισμός του x_s χρησιμοποιώντας το ιδιωτικό κλειδί του υπογράφοντος:

$$x_s = g_s^{-1}(y_s)$$



Σχήμα 8.5: Η δακτυλιοειδής δομή του αλγορίθμου δημιουργίας υπογραφής δακτυλίου.

6. Η υπογραφή δαχτυλιδιού αποτελείται από την ακόλουθη πλειάδα $2n + 1$ τιμών:
 $(P_1, P_2, \dots, P_n; v; x_1, x_2, \dots, x_n)$.

Επαλήθευση Υπογραφών

Η επαλήθευση μιας υπογραφής δακτυλίου περιλαμβάνει τα ακόλουθα τρία βήματα:

1. Εφαρμογή της συνάρτησης καταπακτής δημόσιου κλειδιού σε όλα τα x_i : $y_i = g_i(x_i)$.
2. Υπολογισμός του συμμετρικού κλειδιού $k = H(m)$.
3. Επαλήθευση ότι ισχύει η εξίσωση δακτυλίου $C_{k,v}(y_1, y_2, \dots, y_n) = v$.

8.5 Κρυπτογράφηση Βάσει Ταυτότητας και Χαρακτηριστικών

Στην ενότητα αυτή γίνεται παρουσίαση δύο σχημάτων κρυπτογράφησης που παρουσιάζουν κάποια κοινά στοιχεία, της κρυπτογράφησης βάσει ταυτότητα (Ενότητα 8.5.1) και της κρυπτογράφησης βάσει χαρακτηριστικών (Ενότητα 8.5.2).

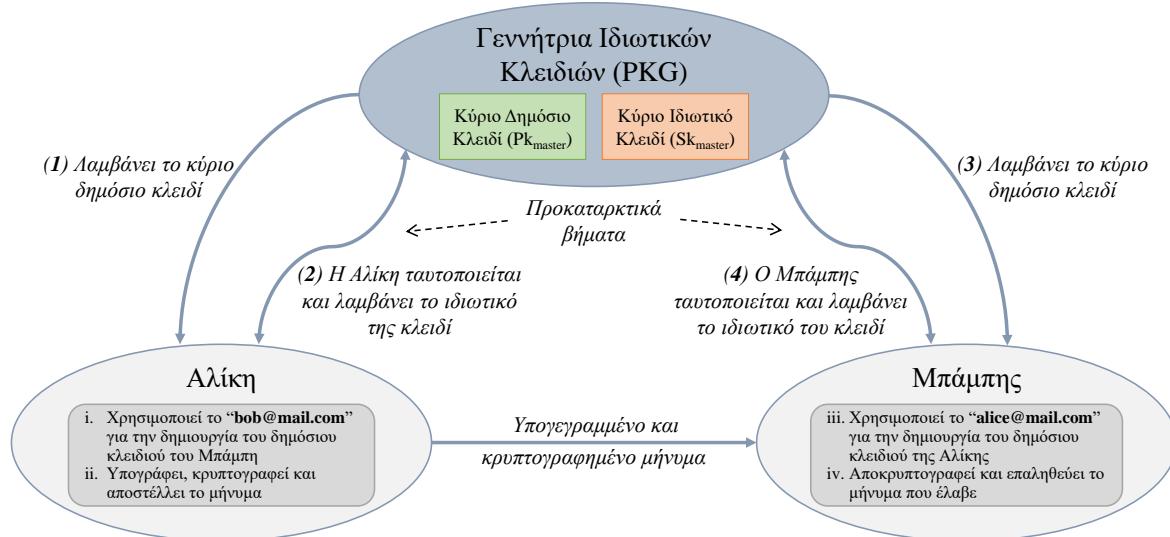
8.5.1 Κρυπτογράφηση Βάσει Ταυτότητας

Η κρυπτογράφηση βάσει ταυτότητας (Identity-Based Encryption – IBE) αποτελεί έναν τύπο κρυπτογράφησης δημοσίου κλειδιού στον οποίο το δημόσιο κλειδί ενός χρήστη αποτελείται από κάποιες μοναδικές πληροφορίες σχετικά με την ταυτότητα του χρήστη (π.χ. η διεύθυνση ήλεκτρονικού ταχυδρομείου). Αυτό σημαίνει ότι ένας αποστολέας που έχει πρόσβαση στις δημόσιες παραμέτρους του συστήματος μπορεί να κρυπτογράφησε ένα μήνυμα χρησιμοποιώντας ως κλειδί, για παράδειγμα, το όνομα ή την διεύθυνση ήλεκτρονικού ταχυδρομείου του παραλήπτη. Αντίστοιχα, ο παραλήπτης αποκτά το κλειδί αποκρυπτογράφησης από μια κεντρική αρχή, η οποία πρέπει να είναι έμπιστη καθώς δημιουργεί κλειδιά για κάθε χρήστη του συστήματος.

Η κρυπτογράφηση βάσει ταυτότητας προτάθηκε για πρώτη φορά από τον Shamir το 1984 [63]. Ωστόσο, το αρχικό αυτό σχήμα ήταν σε θέση να δημιουργεί μόνο υπογραφές βάσει ταυτότητας (identity-based signatures). Η κρυπτογράφηση βάσει ταυτότητας παρέμεινε ανοιχτό πρόβλημα για αρκετά χρόνια, μέχρι το 2001 και την πρόταση του σχήματος Boneh–Franklin (BF-IBE) [64] που αποτελεί τον πρώτο πρακτικό αλγόριθμο IBE, ακολουθώντας αργότερα τα σχήματα Sakai–Kasahara (SK-IBE) [65] και Boneh–Boyen (BB-IBE) [66].

Τα συστήματα κρυπτογράφησης βάσει ταυτότητας (βλέπε Σχήμα 8.6) επιτρέπουν σε οποιαδήποτε οντότητα να δημιουργήσει ένα δημόσιο κλειδί από μια γνωστή τιμή ταυτοποίησης, όπως μια συμβολοσειρά ASCII. Αντίστοιχα, μια έμπιστη οντότητα, που ονομάζεται γεννήτρια ιδιωτικών κλειδιών (Private Key Generator – PKG), είναι αυτή που θα δημιουργήσει τα αντίστοιχα ιδιωτικά κλειδιά. Για να λειτουργήσει αυτό το σύστημα κρυπτογράφησης, η γεννήτρια ιδιωτικών κλειδιών δημοσιεύει πρώτα ένα κύριο δημόσιο κλειδί και κρατάει κρυφό το αντίστοιχο κύριο ιδιωτικό κλειδί (αναφέρεται ως κύριο κλειδί). Δεδομένου του κύριου δημόσιου κλειδιού, οποιαδήποτε οντότητα μπορεί να υπολογίσει ένα δημόσιο κλειδί που αντιστοιχεί στην ταυτότητα του χρήστη συνδυάζοντας το κύριο δημόσιο κλειδί με την τιμή ταυτοποίησης. Για να λάβει το αντίστοιχο ιδιωτικό κλειδί, η οντότητα που εξουσιοδοτείται να χρησιμοποιήσει το αναγνωριστικό ταυτότητας επικοινωνεί με την γεννήτρια ιδιωτικών κλειδιών, η οποία χρησιμοποιεί το κύριο ιδιωτικό κλειδί για να δημιουργήσει το ιδιωτικό κλειδί για το συγκεκριμένο αναγνωριστικό ταυτότητας.

Ως αποτέλεσμα, οι διάφορες οντότητες του συστήματος μπορούν να κρυπτογραφούν μηνύματα (ή να επαληθεύουν υπογραφές) χωρίς να απαιτείται προηγουμένως διανομή κλειδιών μεταξύ των συμμετεχόντων. Αυτό είναι εξαιρετικά χρήσιμο σε περιπτώσεις όπου η προ-διανομή πιστοποιημένων κλειδιών είναι ακατάλληλη ή ανέφικτη λόγω τεχνικών περιορισμάν. Ωστόσο, για την αποκρυπτογράφηση ή την υπογραφή μηνυμάτων, ο εξουσιοδοτημένος χρήστης πρέπει να λάβει το κατάλληλο ιδιωτικό κλειδί από την γεννήτρια ιδιωτικών κλειδιών. Μια σημαντική επιφύλαξη αυτής της προσέγγισης είναι ότι η γεννήτρια ιδιωτικών κλειδιών πρέπει να είναι έμπιστη, καθώς είναι ικανή να δημιουργήσει το ιδιωτικό κλειδί οποιουδήποτε χρήστη και μπορεί επομένως να αποκρυπτογραφήσει (ή να υπογράψει) μηνύματα χωρίς την απαραίτητη εξουσιοδότηση. Επειδή το ιδιωτικό κλειδί οποιουδήποτε χρήστη μπορεί να δημιουργηθεί μέσω της χρήσης ενός μυστικού κλειδιού μιας τρίτης οντότητας, αυτό το σύστημα χαρακτηρίζεται ως ένα σύστημα παρακαταθήκης κλειδιού (key escrow), γνωστό και ως «δίκαιο» κρυπτοσύστημα. Για την αντιμετώπιση αυτού του προβλήματος, έχουν προταθεί μια σειρά από εναλλακτικά συστήματα, όπως η κρυπτογράφηση βάσει πιστοποιητικού (certificate-based encryption) [67], η ασφαλής έκδοση κλειδιού (secure key issuing) [68], και η κρυπτογράφηση χωρίς πιστοποιητικό (certificateless encryption) [69].



Σχήμα 8.6: Επισκόπηση των βημάτων κρυπτογράφησης βάσει ταυτότητας.

8.5.2 Κρυπτογράφηση Βάσει Χαρακτηριστικών

Η κρυπτογράφηση βάσει χαρακτηριστικών (Attribute-Based Encryption – ABE) είναι ένας τύπος κρυπτογράφησης δημόσιου κλειδιού στον οποίο το μυστικό κλειδί ενός χρήστη και το κρυπτοκείμενο εξαρτώνται από χαρακτηριστικά, όπως για παράδειγμα η χώρα διαμονής ή/και το είδος συνδρομής που διαθέτουν. Σε

ένα τέτοιο σύστημα, η αποκρυπτογράφηση ενός κρυπτοκειμένου είναι δυνατή μόνο εάν το σύνολο των χαρακτηριστικών του κλειδιού ενός χρήστη ταιριάζει με τα χαρακτηριστικά του κρυπτοκειμένου. Μια κρίσιμη συνιστώσα ασφαλείας της κρυπτογράφησης βάσει χαρακτηριστικών είναι η ανθεκτικότητα σε «αθέμιτες συμπράξεις» (collusion-resistance): Ένας αντίταλος που διαθέτει πολλαπλά κλειδιά θα πρέπει να μπορεί να έχει πρόσβαση σε δεδομένα μόνο εάν τουλάχιστον ένα μεμονωμένο κλειδί του παρέχει πρόσβαση.

Η έννοια της κρυπτογράφησης βάσει χαρακτηριστικών προτάθηκε για πρώτη φορά το 2005 από τους Sahai και Waters [70] και ακολούθησαν σύντομα και άλλες προσπάθειες [71]. Μεταγενέστερα, αρκετοί ερευνητές πρότειναν περαιτέρω σχήματα κρυπτογράφησης βάσει χαρακτηριστικών που κάνουν χρήση πολλαπλών αρχών για την δημιουργία από κοινού των ιδιωτικών κλειδιών των χρηστών [72, 73].

Η κρυπτογράφηση βάσει χαρακτηριστικών μπορεί να διαχωριστεί σε δύο διαφορετικούς τύπους κρυπτογράφησης: Στην κρυπτογράφηση βάσει χαρακτηριστικών που βασίζεται σε πολιτική κλειδιών (Key-Policy Attribute-Based Encryption – KP-ABE) [71] και στην κρυπτογράφηση βάσει χαρακτηριστικών που βασίζεται σε πολιτική κρυπτοκειμένων (Ciphertext-Policy Attribute-Based Encryption – CP-ABE) [74]. Στο KP-ABE, τα μυστικά κλειδιά των χρηστών δημιουργούνται με βάση ένα δέντρο πρόσβασης που καθορίζει το πεδίο δικαιωμάτων του ενδιαφερόμενου χρήστη και τα δεδομένα κρυπτογραφούνται σε ένα σύνολο χαρακτηριστικών. Ωστόσο, το CP-ABE χρησιμοποιεί δέντρα πρόσβασης για την κρυπτογράφηση δεδομένων και τα μυστικά κλειδιά των χρηστών δημιουργούνται για ένα σύνολο χαρακτηριστικών.

Αν και η κρυπτογράφηση βάσει χαρακτηριστικών είναι πολύ ισχυρή και ένας πολλά υποσχόμενος μηχανισμός, τα συστήματα αυτά υστερούν όσον αφορά την αποδοτικότητά τους και την μη ύπαρξη μηχανισμού ανάκλησης χαρακτηριστικών. Επιπρόσθετα, σε αυτά συγκαταλέγονται προκλήσεις που αφορούν τον συντονισμό κλειδιών (key coordination), την παρακαταθήκη κλειδιών (key escrow), και την ανάκληση κλειδιών (key revocation).

8.6 Ασκήσεις-Εργασίες

Ασκήσεις

8.6.1 Παρακάτω παρουσιάζονται τα βήματα του πρωτοκόλλου μιας απλής λύσης του γνωστού ασφαλούς υπόλογισμού δύο οντοτήτων «Το Πρόβλημα των Εκατομμυριούχων»:

1. Ο Μπάμπης δημιουργεί *n* ταυτόσημα κουτιά.
2. Επιλέγει έναν τυχαίο αριθμό, τον οποίο τοποθετεί στο κουτί *b* (η θέση του κουτιού αντιστοιχεί στο ποσό που κατέχει, τα κουτιά είναι ταξινομημένα και υποδηλώνουν ένα αύξων ποσό το καθένα, π.χ. 5, 10, 15, 20, ...).
3. Τα υπόλοιπα κουτιά τα γεμίζει επίσης με τυχαίους αριθμούς, οι οποίοι όμως δεν έχουν κάποια ιδιαίτερη σημασία και θεωρούνται αναλώσιμοι.
4. Όλα τα κουτιά αποστέλλονται στην Αλίκη.
5. Η Αλίκη με τη σειρά της τα ανοίγει.
6. Αφήνει το περιεχόμενο στα πρώτα *a* από αυτά αμετάβλητο, ενώ αυξάνει κατά 1 τα υπόλοιπα *n - a* κουτιά (προϋπόθεση είναι επίσης ότι δεν θα αλλάξει την σειρά των κουτιών).
7. Όλα τα κουτιά αποστέλλονται στον Μπάμπη.
8. Αν ο αριθμός του Μπάμπη (αυτός που βρίσκεται στο κουτί *b*) έχει αλλάξει, τότε αυτός είναι πλουσιότερος, ενώ αν έχει παραμείνει ο ίδιος τότε η Αλίκη είναι πλουσιότερη.

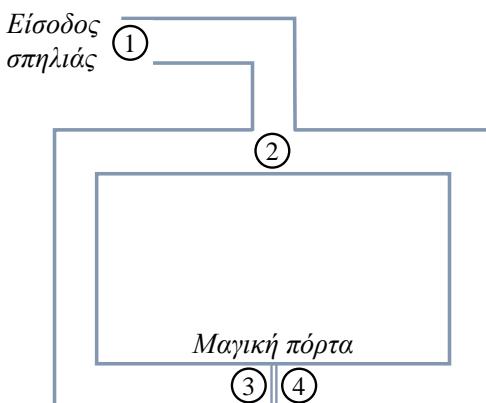
Μπορείτε να εντοπίσετε ποια είναι τα κύρια μειονεκτήματα του παραπάνω απλού πρωτοκόλλου για την επίλυση του προβλήματος των εκατομμυριούχων;

8.6.2 Να υπολογίσετε το αποτέλεσμα των παρακάτω τριών ομοιμορφικών υπολογισμών χρησιμοποιώντας ως x_1 τον αριθμό 20 και ως x_2 τον αριθμό 4:

- (1) Κρυπτοσύστημα RSA: $D(E(x_1) \cdot E(x_2)) = ?$
- (2) Κρυπτοσύστημα Paillier: $D(E(x_1) \cdot E(x_2)) = ?$
- (3) Κρυπτοσύστημα ElGamal: $D(E(x_2) \cdot E(x_2)) = ?$

Όπου E δηλώνει κρυπτογράφηση, το D δηλώνει αποκρυπτογράφηση και το \cdot αποτελεί τη πράξη του γινομένου.

8.6.3 Στην άσκηση αυτή καλείστε να περιγράψετε να βήματα που πρέπει να ακολουθηθούν στο γνωστό παράδειγμα απόδειξης μηδενικής γνώσης «Η Μαγική Πόρτα». Πιο αναλυτικά, στο βάθος μιας σπηλιάς υπάρχει μια μαγική πόρτα που μπορεί να ανοίξει μόνο χρησιμοποιώντας ένα μυστικό κωδικό (βλέπε Σχήμα 8.7). Ποια είναι τα βήματα που πρέπει να ακολουθηθούν ώστε ο Μπάμπης να μπορέσει να πείσει την Αλίκη ότι γνωρίζει τον μυστικό κωδικό, και συνεπώς μπορεί να ανοίγει την μαγική πόρτα, χωρίς να αποκαλύψει στην Αλίκη τον κωδικό;



Σχήμα 8.7: Η θέση της μαγικής πόρτας μεταξύ των σημείων 3 και 4 στην σπηλιά που μπορεί να ανοίξει μόνο με την χρήση ενός μυστικού κλειδιού.

Εργασίες

8.6.1 Σε αυτήν την εργασία θα εξετάσετε την περίπτωση όπου θέλουμε να πραγματοποιήσουμε έναν ομοιμορφικό υπολογισμό κάνοντας χρήση κρυπτοσύστημάτων δημοσίου κλειδιού και πραγματοποιώντας μερική ομοιμορφική κρυπτογράφηση. Συγκεκριμένα, να χρησιμοποιήσετε το κρυπτοσύστημα RSA (πολλαπλασιαστική ιδιότητα) και το κρυπτοσύστημα Paillier (αθροιστική ιδιότητα). Ως είσοδο για το κάθε κρυπτοσύστημα να χρησιμοποιήσετε δύο ακέραιους αριθμούς x_1 και x_2 , και ως έξοδο να υπολογίσετε το αποτέλεσμα της πράξης $D(E(x_1) \cdot E(x_2))$. Στην εργασία αυτή να κάνετε χρήση του εργαλείου [Cryptool 2](#).



8.6.2 Σε αυτήν την εργασία θα δοκιμάσετε διάφορες ομοιμορφικές κρυπτογραφήσεις με χρήση της γλώσσας προγραμματισμού [Java](#) και το περιβάλλον ανάπτυξης [Eclipse IDE for Java Developers](#). Πιο αναλυτικά, θα δοκιμάσετε την μερική ομοιμορφική κρυπτογράφηση του RSA, ElGamal και Paillier. Κάνοντας χρήση του Eclipse Project “[crypto_chap08_HE](#)”, οι δοκιμές που θα πρέπει να γίνουν είναι οι εξής:



- (1) Για να δοκιμάσετε την μερική ομομορφική κρυπτογράφηση του RSA, ElGamal και Paillier εκτελέστε το αρχείο `TestHomomorphicEncryption.java`. Επιπλέον, στον κώδικα που σας δίνεται δοκιμάστε να αλλάξετε τους αριθμούς m_1 και m_2 σε 10 και 5, αντίστοιχα, και δείτε ποια είναι τα αποτέλεσμα αποκρυπτογράφησης των ομομορφικών πράξεων.
- (2) Στο παραπάνω αρχείο Java, αντί να βάλετε την τιμή 5 στο m_2 δοκιμάστε να βάλετε την τιμή -5. Τι θα συμβεί στα αποτελέσματα αποκρυπτογράφησης των ομομορφικών πράξεων για τα τρία κρυπτοσυστήματα;
- (3) Τέλος, και ως συνέχεια του προηγούμενου ερωτήματος, δοκιμάστε να αφαιρέσετε από το αποκρυπτογραφημένο αποτέλεσμα του RSA και ElGamal το μόντουλο τους, δηλ., "... .subtract(pkpRSA.PublicKey); καιsubtract(pkpElGamal.PublicKey.p));", αντίστοιχα. Ποια είναι πλέον τα αποτέλεσματα αποκρυπτογράφησης;

8.6.3 Σε αυτήν την εργασία θα δοκιμάσετε μια απόδειξη μηδενικής γνώσης με χρήση της γλώσσας προγραμματισμού [Java](#) και το περιβάλλον ανάπτυξης [Eclipse IDE for Java Developers](#). Πιο αναλυτικά, θα δοκιμάσετε την απόδειξη μηδενικής γνώσης που μπορεί να αποδείξει ότι ένα κρυπτογραφημένο μήνυμα ανήκει σε ένα δεδομένο σύνολο μηνυμάτων $S = \{m_1, \dots, m_p\}$ χωρίς να αποκαλυφθεί σε καμία περίπτωση το κρυπτογραφημένο μήνυμα. Κάνοντας χρήση του Eclipse Project "[crypto_chap08_ZKP](#)", οι δοκιμές που θα πρέπει να γίνουν είναι οι εξής:



- (1) Για να δοκιμάσετε την απόδειξη μηδενικής γνώσης εκτελέστε το αρχείο `TestZKP.java`. Αναγνωρίζετε σε πιο σημείο κατά κύριο λόγο αυτού το κώδικα εκτελούνται τα βήματα του διαδραστικού πρωτοκόλλου που αναφέρεται στην σελίδα 231 της δημοσίευσης με doi: [10.1007/978-3-642-30436-1_19](https://doi.org/10.1007/978-3-642-30436-1_19).
- (2) Στο παραπάνω αρχείο Java, δοκιμάστε να αλλάξετε το σύνολο τιμών από $S = \{0, 2\}$ σε $S = \{0, 1\}$. Η απόδειξη συνεχίζει να είναι έγκυρη πλέον;
- (3) Τέλος, και ως συνέχεια του προηγούμενου ερωτήματος, δοκιμάστε επιπλέον να αλλάξετε το μήνυμα $m = 2$ σε $m = 1$. Τώρα ποιο είναι πλέον το αποτέλεσμα της απόδειξης;

8.6.4 Σε αυτήν την εργασία θα εξετάσετε την περίπτωση όπου θέλουμε να δημιουργήσουμε τυφλές υπογραφές διασφαλίζοντας ότι ο υπογράφων δεν βλέπει το μήνυμα και υπογράφει το τυφλό μήνυμα. Πιο συγκεκριμένα, να χρησιμοποιήσετε τυφλές υπογραφές με βάση το κρυπτοσύστημα RSA και η διαδικασία της τυφλής υπογραφής να πραγματοποιείται στη σύνοψη του μηνύματος με τον αλγόριθμο σύνοψης SHA-256. Στην εργασία αυτή να κάνετε χρήση του εργαλείου [CrypTool 2](#).



Βιβλιογραφία

- [1] David Evans, Vladimir Kolesnikov, and Mike Rosulek. "A Pragmatic Introduction to Secure Multi-Party Computation". In: *Foundations and Trends® in Privacy and Security* 2.2-3 (2018), pp. 70–246. ISSN: 2474-1558. doi: [10.1561/3300000019](https://doi.org/10.1561/3300000019).
- [2] Andrew C. Yao. "Protocols for Secure Computations". In: *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*. 1982, pp. 160–164. doi: [10.1109/SFCS.1982.38](https://doi.org/10.1109/SFCS.1982.38).
- [3] Dahlia Malkhi, Noam Nisan, Benny Pinkas, and Yaron Sella. "Fairplay – A Secure Two-Party Computation System". In: *Proceedings of the 13th Conference on USENIX Security Symposium*. San Diego, CA: USENIX Association, 2004, p. 16.

- [4] Andrew Chi-Chih Yao. "How to Generate and Exchange Secrets". In: *27th Annual Symposium on Foundations of Computer Science (SFCS 1986)*. IEEE, 1986, pp. 162–167. doi: 10.1109/SFCS.1986.25.
- [5] Michael O. Rabin. *How to Exchange Secrets with Oblivious Transfer*. Tech. rep. TR-81. Aiken Computation Lab, Harvard University, 1981.
- [6] Johannes Blömer, Martin Otto, and Jean-Pierre Seifert. "A New CRT-RSA Algorithm Secure against Bellcore Attacks". In: *Proceedings of the 10th ACM Conference on Computer and Communications Security*. CCS '03. Washington D.C., USA: ACM, 2003, pp. 311–320. ISBN: 1581137389. doi: 10.1145/948109.948151.
- [7] Ebrahim M. Songhori, Siam U. Hussain, Ahmad-Reza Sadeghi, Thomas Schneider, and Farnaz Koushanfar. "TinyGarble: Highly Compressed and Scalable Sequential Garbled Circuits". In: *IEEE Symposium on Security and Privacy*. IEEE, 2015, pp. 411–428. doi: 10.1109/SP.2015.32.
- [8] O. Goldreich, S. Micali, and A. Wigderson. "How to Play Any Mental Game, or a Completeness Theorem for Protocols with Honest Majority". In: *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*. STOC '87. New York, New York, USA: Association for Computing Machinery, 1987, pp. 218–229. ISBN: 0897912217. doi: 10.1145/28395.28420.
- [9] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. "Completeness Theorems for Non-Cryptographic Fault-Tolerant Distributed Computation". In: *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*. STOC '88. Chicago, Illinois, USA: Association for Computing Machinery, 1988, pp. 1–10. ISBN: 0897912640. doi: 10.1145/62212.62213.
- [10] D. Beaver, S. Micali, and P. Rogaway. "The Round Complexity of Secure Protocols". In: *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing*. STOC '90. Baltimore, Maryland, USA: Association for Computing Machinery, 1990, pp. 503–513. ISBN: 0897913612. doi: 10.1145/100216.100287.
- [11] Vladimir Kolesnikov. "Gate Evaluation Secret Sharing and Secure One-Round Two-Party Computation". In: *Advances in Cryptology - ASIACRYPT 2005*. Ed. by Bimal Roy. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 136–155. ISBN: 978-3-540-32267-2. doi: 10.1007/11593447_8.
- [12] Frederik Armknecht et al. *A Guide to Fully Homomorphic Encryption*. Cryptology ePrint Archive, Report 2015/1192. <https://ia.cr/2015/1192>. 2015.
- [13] Craig Gentry and Shai Halevi. "Implementing Gentry's Fully-Homomorphic Encryption Scheme". In: *Advances in Cryptology – EUROCRYPT 2011*. Ed. by Kenneth G. Paterson. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 129–148. ISBN: 978-3-642-20465-4. doi: 10.1007/978-3-642-20465-4_9.
- [14] Mark A. Will and Ryan K.L. Ko. "A Guide to Homomorphic Encryption". In: *The Cloud Security Ecosystem*. Ed. by Ryan Ko and Kim-Kwang Raymond Choo. Boston: Syngress, 2015. Chap. 5, pp. 101–127. ISBN: 978-0-12-801595-7. doi: 10.1016/B978-0-12-801595-7.00005-7.
- [15] Ronald L. Rivest, Len Adleman, and Michael L. Dertouzos. "On data banks and privacy homomorphisms". In: *Foundations of Secure Computation* 4.11 (1978), pp. 169–180.
- [16] G.R. Blakley and Catherine Meadows. "A Database Encryption Scheme Which Allows the Computation of Statistics Using Encrypted Data". In: *IEEE Symposium on Security and Privacy*. IEEE, 1985, pp. 116–116. doi: 10.1109/SP.1985.10024.
- [17] Niv Ahituv, Yeheskel Lapid, and Seev Neumann. "Processing Encrypted Data". In: *Commun. ACM* 30.9 (Sept. 1987), pp. 777–780. ISSN: 0001-0782. doi: 10.1145/30401.30404.

- [18] Ernest F. Brickell and Yacov Yacobi. "On Privacy Homomorphisms". In: *Advances in Cryptology — EUROCRYPT' 87*. Ed. by David Chaum and Wyn L. Price. Berlin, Heidelberg: Springer Berlin Heidelberg, 1988, pp. 117–125. ISBN: 978-3-540-39118-0. doi: 10.1007/3-540-39118-5_12.
- [19] Josep Domingo I. Ferrer. "A New Privacy Homomorphism and Applications". In: *Information Processing Letters* 60.5 (1996), pp. 277–282. issn: 0020-0190. doi: 10.1016/S0020-0190(96)00170-6.
- [20] David Wagner. "Cryptanalysis of an Algebraic Privacy Homomorphism". In: *Information Security*. Ed. by Colin Boyd and Wenbo Mao. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 234–239. ISBN: 978-3-540-39981-0. doi: 10.1007/10958513_18.
- [21] Frederik Armknecht and Ahmad-Reza Sadeghi. *A New Approach for Algebraically Homomorphic Encryption*. Cryptology ePrint Archive, Report 2008/422. <https://ia.cr/2008/422>. 2008.
- [22] Craig Gentry. "Fully Homomorphic Encryption Using Ideal Lattices". In: *Forty-First Annual ACM Symposium on Theory of Computing*. STOC '09. Bethesda, MD, USA: ACM, 2009, pp. 169–178. ISBN: 9781605585062. doi: 10.1145/1536414.1536440.
- [23] Marten van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. "Fully Homomorphic Encryption over the Integers". In: *Advances in Cryptology – EUROCRYPT 2010*. Ed. by Henri Gilbert. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 24–43. ISBN: 978-3-642-13190-5. doi: 10.1007/978-3-642-13190-5_2.
- [24] Craig Gentry, Amit Sahai, and Brent Waters. "Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based". In: *Advances in Cryptology – CRYPTO 2013*. Ed. by Ran Canetti and Juan A. Garay. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 75–92. ISBN: 978-3-642-40041-4. doi: 10.1007/978-3-642-40041-4_5.
- [25] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. "Homomorphic Encryption for Arithmetic of Approximate Numbers". In: *Advances in Cryptology – ASIACRYPT 2017*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Cham: Springer International Publishing, 2017, pp. 409–437. ISBN: 978-3-319-70694-8. doi: 10.1007/978-3-319-70694-8_15.
- [26] Jung Hee Cheon, Seungwan Hong, and Duhyeong Kim. *Remark on the Security of CKKS Scheme in Practice*. Cryptology ePrint Archive, Report 2020/1581. <https://ia.cr/2020/1581>. 2020.
- [27] Taher Elgamal. "A public key cryptosystem and a signature scheme based on discrete logarithms". In: *IEEE Transactions on Information Theory* 31.4 (1985), pp. 469–472. doi: 10.1109/TIT.1985.1057074.
- [28] Shafi Goldwasser and Silvio Micali. "Probabilistic encryption". In: *Journal of Computer and System Sciences* 28.2 (1984), pp. 270–299. issn: 0022-0000. doi: 10.1016/0022-0000(84)90070-9.
- [29] Josh Benaloh. "Dense probabilistic encryption". In: *Proceedings of the Workshop on Selected Areas of Cryptography*. 1994, pp. 120–128.
- [30] Pascal Paillier. "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes". In: *Advances in Cryptology — EUROCRYPT '99*. Ed. by Jacques Stern. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 223–238. ISBN: 978-3-540-48910-8.
- [31] Daniele Micciancio and Oded Regev. "Lattice-based Cryptography". In: *Post-Quantum Cryptography*. Ed. by Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 147–191. ISBN: 978-3-540-88702-7. doi: 10.1007/978-3-540-88702-7_5.
- [32] Craig Gentry. "A Fully Homomorphic Encryption Scheme". <https://crypto.stanford.edu/craig>. PhD thesis. Stanford University, 2009.

- [33] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. “(Leveled) Fully Homomorphic Encryption without Bootstrapping”. In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. ITCS ’12. Cambridge, Massachusetts: Association for Computing Machinery, 2012, pp. 309–325. ISBN: 9781450311151. DOI: 10.1145/2090236.2090262.
- [34] Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. “On-the-Fly Multiparty Computation on the Cloud via Multikey Fully Homomorphic Encryption”. In: *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*. STOC ’12. New York, New York, USA: ACM, 2012, pp. 1219–1234. ISBN: 9781450312455. DOI: 10.1145/2213977.2214086.
- [35] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. “NTRU: A Ring-based Public Key Cryptosystem”. In: *Algorithmic Number Theory*. Ed. by Joe P. Buhler. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 267–288. ISBN: 978-3-540-69113-6. DOI: 10.1007/BFb0054868.
- [36] Junfeng Fan and Frederik Vercauteren. *Somewhat Practical Fully Homomorphic Encryption*. Cryptology ePrint Archive, Report 2012/144. <https://ia.cr/2012/144>. 2012.
- [37] Zvika Brakerski. “Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP”. In: *Advances in Cryptology – CRYPTO 2012*. Ed. by Reihaneh Safavi-Naini and Ran Canetti. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 868–886. ISBN: 978-3-642-32009-5. DOI: 10.1007/978-3-642-32009-5_50.
- [38] Joppe W. Bos, Kristin Lauter, Jake Loftus, and Michael Naehrig. “Improved Security for a Ring-Based Fully Homomorphic Encryption Scheme”. In: *Cryptography and Coding*. Ed. by Martijn Stam. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 45–64. ISBN: 978-3-642-45239-0. DOI: 10.1007/978-3-642-45239-0_4.
- [39] Martin Albrecht, Shi Bai, and Léo Ducas. “A Subfield Lattice Attack on Overstretched NTRU Assumptions”. In: *Advances in Cryptology – CRYPTO 2016*. Ed. by Matthew Robshaw and Jonathan Katz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 153–178. ISBN: 978-3-662-53018-4. DOI: 10.1007/978-3-662-53018-4_6.
- [40] Jung Hee Cheon, Jinkyung Jeong, and Changmin Lee. “An algorithm for NTRU problems and cryptanalysis of the GGH multilinear map without a low-level encoding of zero”. In: *LMS Journal of Computation and Mathematics* 19.A (2016), pp. 255–266. DOI: 10.1112/S1461157016000371.
- [41] Zvika Brakerski and Vinod Vaikuntanathan. “Lattice-Based FHE as Secure as PKE”. In: *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science*. ITCS ’14. Princeton, New Jersey, USA: ACM, 2014, pp. 1–12. ISBN: 9781450326988. DOI: 10.1145/2554797.2554799.
- [42] Jacob Alperin-Sheriff and Chris Peikert. “Faster Bootstrapping with Polynomial Error”. In: *Advances in Cryptology – CRYPTO 2014*. Ed. by Juan A. Garay and Rosario Gennaro. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 297–314. ISBN: 978-3-662-44371-2. DOI: 10.1007/978-3-662-44371-2_17.
- [43] Léo Ducas and Daniele Micciancio. “FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second”. In: *Advances in Cryptology – EUROCRYPT 2015*. Ed. by Elisabeth Oswald and Marc Fischlin. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 617–640. ISBN: 978-3-662-46800-5. DOI: 10.1007/978-3-662-46800-5_24.
- [44] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. “Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds”. In: *Advances in Cryptology – ASIACRYPT 2016*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 3–33. ISBN: 978-3-662-53887-6. DOI: 10.1007/978-3-662-53887-6_1.

- [45] Nicolas Gama, Malika Izabachène, Phong Q. Nguyen, and Xiang Xie. “Structural Lattice Reduction: Generalized Worst-Case to Average-Case Reductions and Homomorphic Cryptosystems”. In: *Advances in Cryptology – EUROCRYPT 2016*. Ed. by Marc Fischlin and Jean-Sébastien Coron. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 528–558. ISBN: 978-3-662-49896-5. doi: [10.1007/978-3-662-49896-5_19](https://doi.org/10.1007/978-3-662-49896-5_19).
- [46] Baiyu Li and Daniele Micciancio. “On the Security of Homomorphic Encryption on Approximate Numbers”. In: *Advances in Cryptology – EUROCRYPT 2021*. Ed. by Anne Canteaut and François-Xavier Standaert. Cham: Springer International Publishing, 2021, pp. 648–677. ISBN: 978-3-030-77870-5. doi: [10.1007/978-3-030-77870-5_23](https://doi.org/10.1007/978-3-030-77870-5_23).
- [47] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. “The Knowledge Complexity of Interactive Proof Systems”. In: *SIAM Journal on Computing* 18.1 (1989), pp. 186–208. doi: [10.1137/0218012](https://doi.org/10.1137/0218012).
- [48] L Babai. “Trading Group Theory for Randomness”. In: *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*. STOC ’85. Providence, Rhode Island, USA: Association for Computing Machinery, 1985, pp. 421–429. ISBN: 0897911512. doi: [10.1145/22145.22192](https://doi.org/10.1145/22145.22192).
- [49] Feng Li and Bruce McMillin. “Chapter Two - A Survey on Zero-Knowledge Proofs”. In: ed. by Ali Hurson. Vol. 94. *Advances in Computers*. Elsevier, 2014, pp. 25–69. doi: [10.1016/B978-0-12-800161-5.00002-5](https://doi.org/10.1016/B978-0-12-800161-5.00002-5).
- [50] Huixin Wu and Feng Wang. “A survey of noninteractive zero knowledge proof system and its applications”. In: *The Scientific World Journal* 2014 (2014). doi: [10.1155/2014/560484](https://doi.org/10.1155/2014/560484).
- [51] C. P. Schnorr. “Efficient Identification and Signatures for Smart Cards”. In: *Advances in Cryptology — CRYPTO’ 89 Proceedings*. Ed. by Gilles Brassard. New York, NY: Springer New York, 1990, pp. 239–252. ISBN: 978-0-387-34805-6. doi: [10.1007/0-387-34805-0_22](https://doi.org/10.1007/0-387-34805-0_22).
- [52] David Chaum and Torben Pryds Pedersen. “Wallet Databases with Observers”. In: *Advances in Cryptology — CRYPTO’ 92*. Ed. by Ernest F. Brickell. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 89–105. ISBN: 978-3-540-48071-6. doi: [10.1007/3-540-48071-4_7](https://doi.org/10.1007/3-540-48071-4_7).
- [53] David Chaum. “Blind Signatures for Untraceable Payments”. In: *Advances in Cryptology*. Ed. by David Chaum, Ronald L. Rivest, and Alan T. Sherman. Boston, MA: Springer US, 1983, pp. 199–203. ISBN: 978-1-4757-0602-4. doi: [10.1007/978-1-4757-0602-4_18](https://doi.org/10.1007/978-1-4757-0602-4_18).
- [54] Gary Locke and Patrick Gallagher. “Digital signature standard (DSS)”. In: *Federal Information Processing Standards Publications* FIPS PUB 186-3 (2009), pp. 1–119.
- [55] Mihir Bellare, Chanathip Namprempre, David Pointcheval, and Michael Semanko. “The One-More-RSA-Inversion Problems and the Security of Chaum’s Blind Signature Scheme”. In: *Journal of Cryptology* 16.3 (2003), pp. 185–215. doi: [10.1007/s00145-002-0120-1](https://doi.org/10.1007/s00145-002-0120-1).
- [56] David Chaum and Eugène van Heyst. “Group Signatures”. In: *Advances in Cryptology — EUROCRYPT’91*. Ed. by Donald W. Davies. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 257–265. ISBN: 978-3-540-46416-7. doi: [10.1007/3-540-46416-6_22](https://doi.org/10.1007/3-540-46416-6_22).
- [57] Jan Camenisch and Markus Michels. “A Group Signature Scheme with Improved Efficiency (Extended Abstract)”. In: *Advances in Cryptology — ASIACRYPT’98*. Ed. by Kazuo Ohta and Dingyi Pei. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 160–174. ISBN: 978-3-540-49649-6. doi: [10.1007/3-540-49649-1_14](https://doi.org/10.1007/3-540-49649-1_14).
- [58] Giuseppe Ateniese, Jan Camenisch, Marc Joye, and Gene Tsudik. “A Practical and Provably Secure Coalition-Resistant Group Signature Scheme”. In: *Advances in Cryptology — CRYPTO 2000*. Ed. by Mihir Bellare. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 255–270. ISBN: 978-3-540-44598-2. doi: [10.1007/3-540-44598-6_16](https://doi.org/10.1007/3-540-44598-6_16).

- [59] Dan Boneh, Xavier Boyen, and Hovav Shacham. "Short Group Signatures". In: *Advances in Cryptology – CRYPTO 2004*. Ed. by Matt Franklin. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 41–55. ISBN: 978-3-540-28628-8. doi: 10.1007/978-3-540-28628-8_3.
- [60] Ronald L. Rivest, Adi Shamir, and Yael Tauman. "How to Leak a Secret". In: *Advances in Cryptology — ASIACRYPT 2001*. Ed. by Colin Boyd. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 552–565. ISBN: 978-3-540-45682-7. doi: 10.1007/3-540-45682-1_32.
- [61] Ronald L. Rivest, Adi Shamir, and Leonard Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Commun. ACM* 21.2 (Feb. 1978), pp. 120–126. ISSN: 0001-0782. doi: 10.1145/359340.359342.
- [62] Michael O. Rabin. *Digitalized Signatures and Public-Key Functions as Intractable as Factorization*. Tech. rep. MIT/LCS/TR-212. Massachusetts Institute of Technology, Laboratory for Computer Science, 1979.
- [63] Adi Shamir. "Identity-Based Cryptosystems and Signature Schemes". In: *Advances in Cryptology*. Ed. by George Robert Blakley and David Chaum. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 47–53. ISBN: 978-3-540-39568-3. doi: 10.1007/3-540-39568-7_5.
- [64] Dan Boneh and Matt Franklin. "Identity-Based Encryption from the Weil Pairing". In: *Advances in Cryptology — CRYPTO 2001*. Ed. by Joe Kilian. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 213–229. ISBN: 978-3-540-44647-7. doi: 10.1007/3-540-44647-8_13.
- [65] Ryuichi Sakai and Masao Kasahara. *ID based Cryptosystems with Pairing on Elliptic Curve*. Cryptology ePrint Archive, Report 2003/054. <https://ia.cr/2003/054>. 2003.
- [66] Dan Boneh and Xavier Boyen. "Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles". In: *Advances in Cryptology - EUROCRYPT 2004*. Ed. by Christian Cachin and Jan L. Camenisch. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 223–238. ISBN: 978-3-540-24676-3. doi: 10.1007/978-3-540-24676-3_14.
- [67] Craig Gentry. "Certificate-Based Encryption and the Certificate Revocation Problem". In: *Advances in Cryptology — EUROCRYPT 2003*. Ed. by Eli Biham. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 272–293. ISBN: 978-3-540-39200-2. doi: 10.1007/3-540-39200-9_17.
- [68] Byoungcheon Lee et al. "Secure Key Issuing in ID-Based Cryptography". In: *Proceedings of the Second Workshop on Australasian Information Security, Data Mining and Web Intelligence, and Software Internationalisation - Volume 32*. ACSW Frontiers '04. Dunedin, New Zealand: Australian Computer Society, Inc., 2004, pp. 69–74. doi: 10.5555/976440.976449.
- [69] Sattam S. Al-Riyami and Kenneth G. Paterson. "Certificateless Public Key Cryptography". In: *Advances in Cryptology - ASIACRYPT 2003*. Ed. by Chi-Sung Laih. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 452–473. ISBN: 978-3-540-40061-5. doi: 10.1007/978-3-540-40061-5_29.
- [70] Amit Sahai and Brent Waters. "Fuzzy Identity-Based Encryption". In: *Advances in Cryptology – EUROCRYPT 2005*. Ed. by Ronald Cramer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 457–473. ISBN: 978-3-540-32055-5. doi: 10.1007/11426639_27.
- [71] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data". In: *Proceedings of the 13th ACM Conference on Computer and Communications Security*. CCS '06. Alexandria, Virginia, USA: Association for Computing Machinery, 2006, pp. 89–98. ISBN: 1595935185. doi: 10.1145/1180405.1180418.

- [72] Melissa Chase and Sherman S.M. Chow. "Improving Privacy and Security in Multi-Authority Attribute-Based Encryption". In: *Proceedings of the 16th ACM Conference on Computer and Communications Security*. CCS '09. Chicago, Illinois, USA: Association for Computing Machinery, 2009, pp. 121–130. ISBN: 9781605588940. doi: 10.1145/1653662.1653678.
- [73] Taeho Jung, Xiang-Yang Li, Zhiguo Wan, and Meng Wan. "Control Cloud Data Access Privilege and Anonymity With Fully Anonymous Attribute-Based Encryption". In: *IEEE Transactions on Information Forensics and Security* 10.1 (2015), pp. 190–199. doi: 10.1109/TIFS.2014.2368352.
- [74] John Bethencourt, Amit Sahai, and Brent Waters. "Ciphertext-Policy Attribute-Based Encryption". In: *IEEE Symposium on Security and Privacy (SP '07)*. IEEE, 2007, pp. 321–334. doi: 10.1109/SP.2007.11.

Μέρος III

ΕΦΑΡΜΟΓΕΣ

ΚΕΦΑΛΑΙΟ 9

ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ ΣΕ ΚΑΤΑΣΤΑΣΗ ΗΡΕΜΙΑΣ

Περίληψη

Η κρυπτογράφηση δεδομένων σε κατάσταση ηρεμίας μπορεί να αφορά τόσο δεδομένα που αποθηκεύονται σε μια βάση δεδομένων, όσο και δεδομένα που αποθηκεύονται σε επίπεδο δίσκου ή ως αρχεία σε επίπεδο συστήματος αρχείων (file system). Το παρόν κεφάλαιο επικεντρώνεται σε όλες αυτές τις διαφορετικές μορφές δεδομένων παρουσιάζοντας μια πληθώρα προσεγγίσεων που υπάρχουν στον χώρο. Πιο αναλυτικά, στην Ενότητα 9.1 γίνεται μια εισαγωγή στα δεδομένα σε κατάσταση ηρεμίας και επισημαίνεται η διαφορά τους από τα δεδομένα σε χρήση και τα δεδομένα κατά την μεταφορά. Στην συνέχεια, η Ενότητα 9.2 επικεντρώνεται σε προσεγγίσεις που αφορούν στην πλήρη κρυπτογράφηση δεδομένων σε επίπεδο δίσκου, ενώ στην Ενότητα 9.3 παρουσιάζονται προσεγγίσεις για την κρυπτογράφηση δεδομένων σε επίπεδο συστήματος αρχείων. Τέλος, στην Ενότητα 9.4 αναφέρονται προσεγγίσεις για την κρυπτογράφηση δεδομένων σε επίπεδο μιας βάσης δεδομένων, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών αρχών της συμμετρικής κρυπτογραφίας (Κεφάλαιο 1) και της κρυπτογραφίας δημοσίου κλειδιού (Κεφάλαιο 2).

9.1 Δεδομένα σε Κατάσταση Ηρεμίας

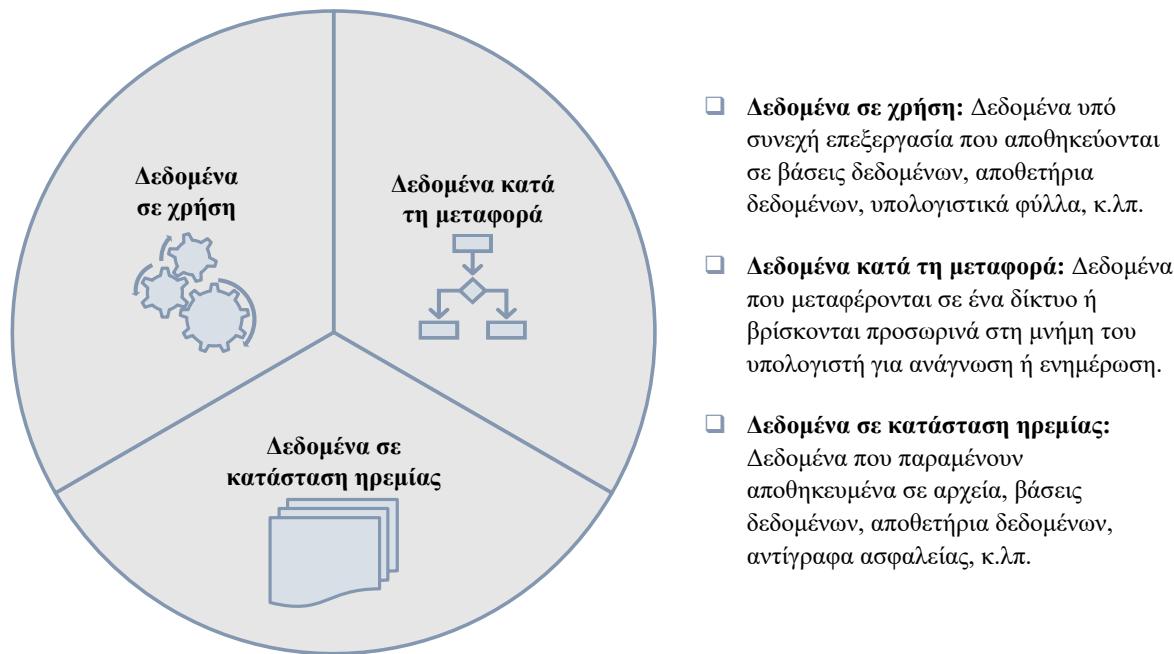
Στην πληροφορική όταν γίνεται αναφορά σε δεδομένα σε κατάσταση ηρεμίας (data at rest), αυτό αφορά δεδομένα υπολογιστή που παραμένουν αποθηκευμένα σε οποιαδήποτε ψηφιακή μορφή (π.χ. στον σκληρό δίσκο, σε υπηρεσίες φιλοξενίας αρχείων, σε βάσεις δεδομένων, στο Νέφος (Cloud), σε αντίγραφα ασφαλείας, σε φορητές συσκευές, κ.λπ.). Επιπλέον, τα δεδομένα σε κατάσταση ηρεμίας περιλαμβάνουν τόσο δομημένα όσο και αδόμητα δεδομένα. Αυτός ο τύπος δεδομένων υπόκειται σε διάφορες κακόβουλες απειλές που αποσκοπούν στο να αποκτήσουν πρόσβαση στα ψηφιακά δεδομένα ή στη φυσική κλοπή των μέσων αποθήκευσης τους. Για να αποφευχθεί η πρόσβαση, η τροποποίηση ή η κλοπή αυτών των δεδομένων, χρησιμοποιούνται συχνά διάφορα μέτρα προστασίας, όπως η προστασία με χρήση κωδικού πρόσβασης, η κρυπτογράφηση δεδομένων ή συνδυασμό και των δύο. Οι επιλογές ασφαλείας που χρησιμοποιούνται για αυτόν τον τύπο δεδομένων ανα-

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

φέρονται συχνά ως προστασία δεδομένων σε κατάσταση ηρεμίας (Data At Rest Protection – DARP).

Τα δεδομένα σε κατάσταση ηρεμίας χρησιμοποιούνται συμπληρωματικά των όρων «δεδομένα σε χρήση» (data in use) και «δεδομένα κατά τη μεταφορά» (data in transit), και όλα μαζί ορίζουν τις τρεις καταστάσεις των ψηφιακών δεδομένων (βλέπε Σχήμα 9.1). Σε αυτό το κεφάλαιο επικεντρωνόμαστε στα δεδομένα σε κατάσταση ηρεμίας, ενώ στα δεδομένα κατά την μεταφορά γίνεται εκτεταμένη αναφορά στο Κεφάλαιο 10. Όσον αφορά τα δεδομένα σε χρήση, στο Κεφάλαιο 8 έγινε εκτεταμένη αναφορά σε μηχανισμούς ενίσχυσης του απορρήτου που αξιοποιούν τέτοια δεδομένα με ταυτόχρονη προστασία τους.



Σχήμα 9.1: Οι τρεις καταστάσεις των ψηφιακών δεδομένων.

Λόγω της φύσης τους, τα δεδομένα σε κατάσταση ηρεμίας προκαλούν αυξανόμενη ανησυχία για τις επιχειρήσεις, τις κρατικές υπηρεσίες αλλά και διάφορους άλλους φορείς. Οι φορητές συσκευές συχνά υπόκεινται σε συγκεκριμένα πρωτόκολλα ασφαλείας για την προστασία των δεδομένων σε κατάσταση ηρεμίας από μη εξουσιοδοτημένη πρόσβαση, ειδικά όταν αυτές χαθούν ή κλαπούν [1]. Επιπλέον, επικρατεί ολοένα και περισσότερο η άποψη ότι τα συστήματα διαχείρισης βάσεων δεδομένων και οι διακομιστές αρχείων θα πρέπει επίσης να θεωρούνται ότι βρίσκονται σε έναν διαρκή κίνδυνο κυβερνοεπιθέσεων [2]. Πιο συγκεκριμένα, όσο περισσότερο τα δεδομένα παραμένουν αχρησιμοποίητα σε ένα σύστημα αποθήκευσης, τόσο πιο πιθανό είναι να ανακτηθούν από μη εξουσιοδοτημένα άτομα. Για αυτό τον λόγο, είναι πλέον επιτακτική η προστασία των διαφόρων δεδομένων σε κατάσταση ηρεμίας με τεχνικές ασφαλείας, οι οποίες χωρίζονται κυρίως στις ακόλουθες δύο κατηγορίες.

Τεχνικές Κρυπτογράφησης: Η κρυπτογράφηση δεδομένων, η οποία εμποδίζει την ανάγνωση των δεδομένων σε περίπτωση μη εξουσιοδοτημένης πρόσβασης ή κλοπής τους, χρησιμοποιείται συνήθως για την προστασία δεδομένων κατά τη μεταφορά. Ωστόσο, προωθείται όλο και περισσότερο για την προστασία δεδομένων σε κατάσταση ηρεμίας [3]. Η κρυπτογράφηση δεδομένων σε κατάσταση ηρεμίας θα πρέπει να χρησιμοποιεί μόνο ισχυρές μεθόδους κρυπτογράφησης, όπως οι AES και RSA. Τα κρυπτογραφημένα δεδομένα θα πρέπει να παραμένουν κρυπτογραφημένα όταν αποτυγχάνουν οι τεχνικές ελέγχου πρόσβασης, όπως το όνομα χρήστη και ο κωδικός πρόσβασης. Επιπλέον, συνιστάται η χρήση της κρυπτογράφησης σε πολλαπλά επίπεδα. Η κρυπτογραφία μπορεί να εφαρμοστεί τόσο στη βάση δεδομένων που φιλοξενεί τα δεδομένα, αλλά όσο και στη φυσική αποθήκευση (π.χ. σκληροί δίσκοι) όπου είναι αποθηκευμένες οι βάσεις δεδομένων. Επιπρόσθετα, τα κλειδιά κρυπτογράφησης δεδομένων θα πρέπει να ενημερώνονται σε τακτά χρονικά διαστήματα και να

αποθηκεύονται ξεχωριστά από τα ίδια τα δεδομένα. Ο περιοδικός έλεγχος της ασφάλειας των ευαίσθητων δεδομένων θα πρέπει να αποτελεί μέρος της πολιτικής ασφάλειας και να πραγματοποιείται σε προγραμματισμένες χρονικές στιγμές. Σε αυτό το κεφάλαιο επικεντρωνόμαστε κυρίως σε κρυπτογραφικές τεχνικές για την προστασία των δεδομένων σε κατάσταση ηρεμίας τόσο σε επίπεδο δίσκου και συστήματος αρχείων, όσο και σε επίπεδο βάσεων δεδομένων.

Τεχνικές Tokenization: Οι τεχνικές tokenization αποτελούν μια μη μαθηματική προσέγγιση για την προστασία δεδομένων σε κατάσταση ηρεμίας που αντικαθιστά τα ευαίσθητα δεδομένα με μη ευαίσθητα υποκατάστατα, γνωστά ως tokens, τα οποία δεν έχουν κάποια εξωγενή, εκμεταλλεύσιμη σημασία ή αξία. Αυτή η διαδικασία δεν αλλάζει τον τύπο ή το μήκος των δεδομένων, πράγμα που σημαίνει ότι μπορούν να γίνουν αντικείμενα επεξεργασίας από παλαιού τύπου συστήματα, όπως βάσεις δεδομένων, που μπορεί να είναι ευαίσθητη στο μήκος και στον τύπο δεδομένων. Η χρήση tokens απαιτεί σημαντικά λιγότερους υπολογιστικούς πόρους για την επεξεργασία και λιγότερο χώρο αποθήκευσης στις βάσεις δεδομένων από ότι τα κρυπτογραφημένα δεδομένα. Αυτό επιτυγχάνεται διατηρώντας συγκεκριμένα δεδομένα πλήρως ή μερικώς ορατά για επεξεργασία και ανάλυση, ενώ οι ευαίσθητες πληροφορίες διατηρούνται κρυφές. Οι χαμηλότερες απαιτήσεις επεξεργασίας και αποθήκευσης καθιστούν τις τεχνικές tokenization μια ιδανική μέθοδο για την ασφάλεια των δεδομένων σε κατάσταση ηρεμίας σε συστήματα που διαχειρίζονται μεγάλους όγκους δεδομένων. Ωστόσο, στο κεφάλαιο αυτό δεν θα αναφερθούμε περαιτέρω στις τεχνικές tokenization.

9.2 Κρυπτογράφηση Δεδομένων σε Επίπεδο Δίσκου

Η κρυπτογράφηση δεδομένων σε επίπεδο δίσκου είναι μια τεχνολογία που προστατεύει τις πληροφορίες μετατρέποντάς τις σε μια μη αναγνώσιμη μορφή που δεν μπορεί εύκολα να αποκρυπτογραφηθεί από μη εξουσιοδοτημένα άτομα. Η κρυπτογράφηση δίσκου (disk encryption) χρησιμοποιεί, τόσο λογισμικό, όσο και υλικό για την κρυπτογράφηση κάθε bit δεδομένων που βρίσκεται σε έναν δίσκο ή τμήμα (volume) δίσκου, αποτρέποντας την μη εξουσιοδοτημένη πρόσβαση στην αποθήκευση δεδομένων.

Η έκφραση «πλήρης κρυπτογράφηση δίσκου» (Full Disk Encryption – FDE) υποδηλώνει ότι τα πάντα στον δίσκο είναι κρυπτογραφημένα, αλλά το κύριο αρχείο εκκίνησης (Master Boot Record – MBR) ή κάποια παρόμοια περιοχή ενός δίσκου που ξεκινά την διαδικασία φόρτωσης του λειτουργικού συστήματος, δεν είναι κρυπτογραφημένη. Ωστόσο, ορισμένα συστήματα πλήρους κρυπτογράφησης δίσκου που βασίζονται σε υλικό μπορούν πραγματικά να κρυπτογραφήσουν ολόκληρο έναν δίσκο, συμπεριλαμβανομένου και του κύριου αρχείου εκκίνησης.

Η κρυπτογράφηση σε επίπεδο δίσκου χρησιμοποιείται μερικές φορές σε συνδυασμό με την κρυπτογράφηση σε επίπεδο συστήματος αρχείων (βλέπε Ενότητα 9.3) με σκοπό την παροχή μιας πιο ασφαλούς υλοποίησης. Εφόσον η κρυπτογράφηση δίσκου χρησιμοποιεί γενικά το ίδιο κλειδί για την κρυπτογράφηση ολόκληρης της μονάδας δίσκου, όλα τα δεδομένα μπορούν να αποκρυπτογραφηθούν μόνο όταν εκτελείται το σύστημα. Ωστόσο, σε ορισμένες λύσεις κρυπτογράφησης δίσκου χρησιμοποιούνται πολλαπλά κλειδιά για την κρυπτογράφηση διαφορετικών τμημάτων (volumes) του δίσκου. Και στις δυο αυτές προσεγγίσεις, όταν ένας εισβολέας αποκτήσει πρόσβαση κατά την εκτέλεση του υπολογιστή, τότε ο εισβολέας αποκτά πρόσβαση σε όλα τα αρχεία. Αντίθετα, στην κρυπτογράφηση σε επίπεδο συστήματος αρχείων χρησιμοποιούνται διαφορετικά κλειδιά σε διαφορετικά σημεία του δίσκου. Επομένως, ένας εισβολέας δεν μπορεί να εξαγάγει πληροφορίες από φακέλους και αρχεία που παραμένουν ακόμη κρυπτογραφημένα. Σε αντίθεση με την κρυπτογράφηση δίσκου, η κρυπτογράφηση σε επίπεδο συστήματος αρχείων δεν κρυπτογραφεί συνήθως τα μεταδεδομένα του συστήματος αρχείων, όπως τη δομή καταλόγου, τα ονόματα αρχείων, τις χρονικές σημάνσεις τροποποίησης ή τα μεγέθη.

Η πλήρης κρυπτογράφηση δίσκου έχει αρκετά πλεονεκτήματα σε σύγκριση με την κρυπτογράφηση σε επίπεδο αρχείων ή φακέλων, τα οποία συνοψίζονται στα ακόλουθα:

- Σχεδόν τα πάντα, συμπεριλαμβανομένου του χώρου εικονικής μνήμης (virtual memory ή swap mem-

ory) και των προσωρινών αρχείων είναι κρυπτογραφημένα. Η κρυπτογράφηση αυτών των αρχείων είναι σημαντική, καθώς μπορούν να αποκαλύψουν σημαντικά εμπιστευτικά δεδομένα. Ωστόσο, σε λύσεις λογισμικού, ο κώδικας εκκίνησης δεν μπορεί να κρυπτογραφηθεί. Για παράδειγμα, το BitLocker (βλέπε Ενότητα 9.2.2) αφήνει έναν μη κρυπτογραφημένο τμήμα (volume) για την εκκίνηση, ενώ το τμήμα που περιέχει το λειτουργικό σύστημα είναι πλήρως κρυπτογραφημένο.

- Με την πλήρη κρυπτογράφηση δίσκου, η απόφαση για το ποια μεμονωμένα αρχεία θα πρέπει να κρυπτογραφηθούν δεν αφήνεται στη διακριτική ευχέρεια των χρηστών. Αυτό είναι σημαντικό για καταστάσεις στις οποίες οι χρήστες μπορεί να μην θέλουν ή μπορεί να ξεχάσουν να κρυπτογραφήσουν ευαίσθητα αρχεία.
- Η άμεση καταστροφή δεδομένων, όπως η απλή διαγραφή των κρυπτογραφικών κλειδιών (γνωστή και ως crypto-shredding), καθιστά τα κρυπτογραφημένα δεδομένα άχρηστα. Ωστόσο, εάν υπάρχει ανησυχία για την ασφάλεια έναντι πιθανών μελλοντικών επιθέσεων, συνιστάται καθαρισμός ή φυσική καταστροφή του μέσου αποθήκευσης.

Διαφανής Κρυπτογράφηση: Η διαφανής κρυπτογράφηση (transparent encryption), γνωστή και ως κρυπτογράφηση σε πραγματικό χρόνο και on-the-fly (On-The-Fly Encryption – OTFE), είναι μια μέθοδος που χρησιμοποιείται από κάποια λογισμικά κρυπτογράφησης δίσκου (όπως του TrueCrypt [4]). Ο όρος «διαφανής» αναφέρεται στο γεγονός ότι τα δεδομένα κρυπτογραφούνται ή αποκρυπτογραφούνται αυτόματα καθώς φορτώνονται ή αποθηκεύονται.

Κάνοντας χρήση της διαφανούς κρυπτογράφησης, τα αρχεία είναι προσβάσιμα αιμέσως μετά την παροχή του κλειδιού και ολόκληρος το τμήμα (volume) συνήθως εμφανίζεται σαν να ήταν μια φυσική μονάδα δίσκου, καθιστώντας τα αρχεία εξίσου προσβάσιμα όπως τα μη κρυπτογραφημένα. Κανένα από τα δεδομένα που είναι αποθηκευμένα σε ένα κρυπτογραφημένο τμήμα δεν μπορούν να διαβαστούν (αποκρυπτογραφηθούν) χωρίς τη χρήση του σωστού κωδικού πρόσβασης ή των σωστών κλειδιών κρυπτογράφησης. Επιπλέον, ολόκληρο το σύστημα αρχείων μέσα στο τμήμα του δίσκου είναι κρυπτογραφημένο, συμπεριλαμβανομένων των ονομάτων των αρχείων, των ονομάτων των φακέλων και άλλων μεταδεδομένων.

Μονάδα Αξιόπιστης Πλατφόρμας (TPM): Η Μονάδα Αξιόπιστης Πλατφόρμας (Trusted Platform Module – TPM) αποτελεί έναν ασφαλή κρυπτοεπεξεργαστή που βρίσκεται ενσωματωμένος στη μητρική πλακέτα και μπορεί να χρησιμοποιηθεί για την επαλήθευση της ταυτότητας μιας συσκευής. Δεδομένου ότι κάθε τσιπ TPM είναι μοναδικό για μια συγκεκριμένη συσκευή, μπορεί να πραγματοποιεί την αυθεντικοποίηση της συσκευής. Επιπλέον, μπορεί να χρησιμοποιηθεί για την επαλήθευση ότι το σύστημα που ζητά πρόσβαση είναι το αναμενόμενο σύστημα [5].

Το TPM υποστηρίζεται από έναν περιορισμένο αριθμό λύσεων κρυπτογράφησης δίσκου. Αυτές οι υλοποιήσεις περικλείουν το κλειδί αποκρυπτογράφησης κάνοντας χρήση του TPM, συνδέοντας έτσι τη μονάδα δίσκου (HDD) σε μια συγκεκριμένη συσκευή υπολογιστή. Εάν ο δίσκος αφαιρεθεί από τη συγκεκριμένη συσκευή και τοποθετηθεί σε άλλη, η διαδικασία αποκρυπτογράφησης θα αποτύχει. Ωστόσο, η ανάκτηση είναι δυνατή με χρήση κωδικού ή διακριτικού (token) αποκρυπτογράφησης.

Αν και αυτό έχει το πλεονέκτημα ότι ο δίσκος δεν μπορεί να αφαιρεθεί από τη συσκευή, μπορεί να δημιουργήσει ωστόσο ένα σημαντικό σημείο αστοχίας στην κρυπτογράφηση. Για παράδειγμα, εάν συμβεί κάτι με το TPM ή τη μητρική πλακέτα, ένας χρήστης δεν θα έχει πρόσβαση στα δεδομένα συνδέοντας τη μονάδα σκληρού δίσκου σε άλλον υπολογιστή, εκτός και εάν αυτός ο χρήστης διαθέτει ξεχωριστό κλειδί ανάκτησης.

Το Πρόβλημα του Κλειδιού Εκκίνησης: Ένα σημαντικό ζήτημα που πρέπει να αντιμετωπιστεί στην πλήρη κρυπτογράφηση δίσκου είναι ότι τα μπλοκ στα οποία είναι αποθηκευμένο το λειτουργικό σύστημα πρέπει να αποκρυπτογραφηθούν πριν ξεκινήσει η εκκίνηση του λειτουργικού συστήματος, πράγμα που σημαίνει ότι το κλειδί πρέπει να είναι διαθέσιμο προτού υπάρξει διεπαφή με τον χρήστη για να του ζητηθεί ο κωδικός πρόσβασης. Οι περισσότερες λύσεις πλήρους κρυπτογράφησης δίσκου χρησιμοποιούν πριν από την εκκίνηση τον έλεγχο ταυτότητας φορτώνοντας ένα μικρό, εξαιρετικά ασφαλές λειτουργικό σύστημα το οποίο είναι αυστηρά

κλειδωμένο σε σχέση με τις μεταβλητές συστήματος για να ελέγξει την ακεραιότητα του πυρήνα προεκκίνησης (pre-boot). Ορισμένες υλοποιήσεις, όπως το BitLocker (βλέπε Ενότητα 9.2.2), μπορούν να κάνουν χρήση υλικού, όπως μιας μονάδας αξιόπιστης πλατφόρμας (TPM), για να διασφαλίσουν την ακεραιότητα του περιβάλλοντος εκκίνησης και, ως εκ τούτου, να αποτρέψουν επιθέσεις που στοχεύουν στην αντικατάσταση του boot loader με μια τροποποιημένη έκδοση. Αυτό διασφαλίζει ότι η αυθεντικοποίηση μπορεί να πραγματοποιηθεί σε ελεγχόμενο περιβάλλον χωρίς τη δυνατότητα χρήσης κάποιου εργαλείου για την ανατροπή της αποκρυπτογράφησης πριν από την εκκίνηση.

Με ένα περιβάλλον ελέγχου ταυτότητας πριν από την εκκίνηση, το κλειδί που χρησιμοποιείται για την κρυπτογράφηση των δεδομένων δεν αποκρυπτογραφείται μέχρι να εισαχθεί ένα εξωτερικό κλειδί στο σύστημα.

Οι λύσεις που χρησιμοποιούνται για την αποθήκευση του εξωτερικού κλειδιού είναι οι εξής:

- Όνομα χρήστη και κωδικός πρόσβασης
- Χρήση έξυπνης κάρτας (smartcard) σε συνδυασμό με κάποιο PIN
- Χρήση βιομετρικής μεθόδου ελέγχου ταυτότητας, όπως δακτυλικό αποτύπωμα
- Χρήση μια συσκευής (dongle) για την αποθήκευση του κλειδιού, με την προϋπόθεση ότι ο χρήστης δεν θα επιτρέψει την κλοπή της συσκευής μαζί με τον υπολογιστή ή ότι η συσκευή είναι επίσης κρυπτογραφημένη
- Χρήση ενός προγράμματος οδήγησης κατά την εκκίνηση που να ζητάει τον κωδικό πρόσβασης από τον χρήστη
- Χρήση του δικτύου για την ανάκτηση του κλειδιού, για παράδειγμα ως μέρος μιας εκκίνησης PXE (Preboot eXecution Environment) [6]
- Χρήση TPM για την αποθήκευση του κλειδιού αποκρυπτογράφησης, αποτρέποντας μη εξουσιοδοτημένη πρόσβαση στο κλειδί αποκρυπτογράφησης ή την φόρτωση τροποποιημένου boot loader
- Συνδυασμός όλων των παραπάνω

Όλες αυτές οι λύσεις για την αποθήκευση του εξωτερικού κλειδιού παρέχουν διάφορους βαθμούς ασφαλειας, ωστόσο είναι καλύτερες από την περίπτωση ενός μη κρυπτογραφημένου δίσκου.

Ζητήματα Ασφαλείας: Τα περισσότερα σχήματα πλήρους κρυπτογράφησης δίσκου είναι ευάλωτα σε μια επίθεση ψυχρής εκκίνησης (cold boot), όπου τα κλειδιά κρυπτογράφησης μπορούν να κλαπούν με ψυχρή εκκίνηση ενός μηχανήματος όπου εκτελείται ήδη το λειτουργικό σύστημα και ανακτώντας τα περιεχόμενα της μνήμης RAM πριν εξαφανιστούν τα δεδομένα. Η επίθεση αυτή βασίζεται στην ιδιότητα διατήρησης των δεδομένων της μνήμης του υπολογιστή, σύμφωνα με την οποία τα bit δεδομένων μπορεί να χρειαστούν έως και αρκετά λεπτά για να εξαλειφθούν μετά την διακοπή της τροφοδοσίας [7]. Ακόμη και μια μονάδα αξιόπιστης πλατφόρμας (TPM) δεν είναι αποτελεσματική σε αυτή την επίθεση, καθώς το λειτουργικό σύστημα πρέπει να κρατά τα κλειδιά αποκρυπτογράφησης στη μνήμη για να έχει πρόσβαση στο δίσκο.

Η πλήρης κρυπτογράφηση δίσκου είναι επίσης ευάλωτη όταν ένας υπολογιστής κλαπεί ενώ βρίσκεται σε αναστολή λειτουργίας. Καθώς η αφύπνιση δεν περιλαμβάνει την διαδικασία εκκίνησης του BIOS, συνήθως αυτό έχει ως αποτέλεσμα να μην ζητηθεί ξανά ο κωδικός πρόσβασης της πλήρους κρυπτογράφησης δίσκου. Αντίθετα, η αδρανοποίηση «περνά» μέσα από τη διαδικασία εκκίνησης του BIOS και επομένως είναι ασφαλής.

Όλα τα συστήματα κρυπτογράφησης που βασίζονται σε λογισμικό είναι ευάλωτα σε διάφορες επιθέσεις πλευρικού καναλιού (side channel), όπως η ακουστική κρυπτανάλυση (acoustic cryptanalysis) και οι key-loggers υλικού. Αντίθετα, οι αυτο-κρυπτογραφούμενες μονάδες δίσκου (self-encrypting drives) δεν είναι ευάλωτες σε αυτές τις επιθέσεις, καθώς το κλειδί κρυπτογράφησης υλικού δεν φεύγει ποτέ από τον ελεγκτή του δίσκου.

Επίσης, τα περισσότερα σχήματα πλήρους κρυπτογράφησης δίσκου δεν παρέχουν προστασία από την παραποίηση δεδομένων (ή την σιωπηρή καταστροφή δεδομένων, π.χ. bit rot [8]). Αυτό σημαίνει ότι παρέχουν μόνο ιδιωτικότητα, αλλά όχι ακεραιότητα. Οι τρόποι λειτουργίας που βασίζονται σε κρυπτογράφηση μπλοκ και χρησιμοποιούνται για την πλήρη κρυπτογράφηση δίσκου δεν παρέχουν αυθεντικοποιημένη κρυπτογράφηση (authenticated encryption) λόγω των ανησυχιών που υπάρχουν σχετικά με την επιβάρυνση αποθήκευσης που απαιτείται για τις ετικέτες αυθεντικοποίησης. Έτσι, εάν γίνει παραποίηση στα δεδομένα του δίσκου, τα δεδομένα που θα αποκρυπτογραφηθούν θα είναι παραποιημένα τυχαία δεδομένα, ελπίζοντας ότι τα σφάλματα που θα προκύψουν θα είναι ανάλογα με τα δεδομένα που έχουν παραποιηθεί (π.χ. για την περίπτωση των μεταδεδομένων του λειτουργικού συστήματος θα επηρεαστεί το σύστημα αρχείων, ενώ για την περίπτωση των δεδομένων ενός αρχείου θα επηρεαστεί το αντίστοιχο πρόγραμμα που θα επεξεργαζόταν το αρχείο). Ένας από τους τρόπους για να μετριαστούν αυτές οι ανησυχίες σχετικά με την παραποίηση δεδομένων, είναι η χρήση συστημάτων αρχείων με πλήρη έλεγχο ακεραιότητας δεδομένων μέσω αθροισμάτων ελέγχου (checksums), όπως το BTRFS [9] και το ZFS [10], πέρα από την πλήρη κρυπτογράφηση δίσκου.

9.2.1 Μέθοδοι Κρυπτογράφησης Δίσκου

Οι μέθοδοι κρυπτογράφησης δίσκου στοχεύουν στην παροχή τριών διακριτών ιδιοτήτων:

1. Τα δεδομένα στο δίσκο πρέπει να παραμένουν εμπιστευτικά.
2. Η ανάκτηση και η αποθήκευση δεδομένων θα πρέπει να είναι γρήγορες λειτουργίες, ανεξάρτητα από το που αποθηκεύονται τα δεδομένα στο δίσκο.
3. Η μέθοδος κρυπτογράφησης δεν πρέπει να σπαταλά χώρο στο δίσκο (δηλ., ο χώρος αποθήκευσης που απαιτείται για τα κρυπτογραφημένα δεδομένα δεν πρέπει να είναι σημαντικά μεγαλύτερος από τα ίδια τα δεδομένα χωρίς κρυπτογράφηση).

Η πρώτη ιδιότητα απαιτεί τον ορισμό ενός αντίπαλου (adversary) για τον οποίο τα δεδομένα τηρούνται εμπιστευτικά. Οι ισχυρότεροι αντίπαλοι που έχουν μελετηθεί στην περιοχή της κρυπτογράφησης δίσκων διαθέτουν τις ακόλουθες ικανότητες:

- Μπορούν να διαβάσουν τα πρωτογενή περιεχόμενα του δίσκου ανά πάσα στιγμή.
- Μπορούν να ζητήσουν από το δίσκο να κρυπτογραφήσει και να αποθηκεύσει αυθαίρετα αρχεία της επιλογής τους.
- Μπορούν να τροποποιήσουν αχρησιμοποίητους τομείς (sectors) του δίσκου και στη συνέχεια να ζητήσουν την αποκρυπτογράφησή τους.

Μια μέθοδος παρέχει καλή εμπιστευτικότητα, εάν και μόνο εάν, η μόνη πληροφορία που μπορεί να προσδιορίσει ένας αντίπαλος με την πάροδο του χρόνου είναι εάν τα δεδομένα σε έναν τομέα έχουν αλλάξει ή όχι από την τελευταία φορά που τα παρατήρησε.

Η δεύτερη ιδιότητα απαιτεί τη διαίρεση του δίσκου σε διάφορους τομείς, συνήθως μεγέθους 512 bytes (ή 4096 bits), οι οποίοι κρυπτογραφούνται και αποκρυπτογραφούνται ανεξάρτητα ο ένας από τον άλλο. Εάν τα δεδομένα θέλουμε να παραμείνουν εμπιστευτικά, η μέθοδος κρυπτογράφησης πρέπει να έχει μη συγκρίσιμη έξοδο και δεν πρέπει να γίνεται επεξεργασία δύο τομέων με τον ίδιο ακριβώς τρόπο. Διαφορετικά, ο αντίπαλος θα μπορούσε να αποκρυπτογραφήσει οποιονδήποτε τομέα του δίσκου αντιγράφοντάς τον σε έναν αχρησιμοπόιητο τομέα του δίσκου και ζητώντας την αποκρυπτογράφηση του.

Η τρίτη ιδιότητα απαγορεύει έμμεσα τη χρήση κρυπτογράφησης ροής, καθώς οι αλγόριθμοι κρυπτογράφησης ροής απαιτούν, για την ασφάλειά τους, να μην χρησιμοποιείται η ίδια αρχική κατάσταση δύο φορές (κάτι που θα συνέβαινε εάν ένας τομέας ενημερώνεται με διαφορετικά δεδομένα). Επομένως, αυτό θα απαιτούσε μια μέθοδο κρυπτογράφησης για την αποθήκευση ξεχωριστών αρχικών καταστάσεων για κάθε τομέα στο δίσκο, που φαινομενικά προϋποθέτει σπατάλη αποθηκευτικού χώρου. Μια εναλλακτική λύση είναι

η χρήση κρυπτογράφησης μπλοκ που περιορίζεται σε ένα συγκεκριμένο μέγεθος μπλοκ (συνήθως 128 ή 256 bits). Εξαιτίας αυτού, η κρυπτογράφηση δίσκου μελετά κυρίως τρόπους λειτουργίας (operation modes), οι οποίοι επεκτείνουν το μήκος του μπλοκ κρυπτογράφησης για να καλύψει έναν ολόκληρο τομέα δίσκου. Όσα έχουν αναφερθεί μέχρι στιγμής καθιστούν ακατάλληλους πολλούς γνωστούς τρόπους λειτουργίας (βλέπε Ενότητα 1.4), όπως η λειτουργία ECB η οποία δεν μπορεί να τροποποιηθεί και λειτουργίες που μετατρέπουν την κρυπτογράφηση μπλοκ σε κρυπτογράφηση ροής, όπως η λειτουργία CTR.

Φυσικά, αυτές οι τρεις ιδιότητες δεν παρέχουν καμία διασφάλιση της ακεραιότητας του δίσκου. Δηλαδή, δεν διασφαλίζουν αν κάποιος αντίταλος έχει τροποποιήσει τα κρυπτοκείμενα. Εν μέρει, αυτό συμβαίνει επειδή η απόλυτη διασφάλιση της ακεραιότητας του δίσκου είναι αδύνατη, γιατί πολύ απλά ένας αντίταλος θα μπορούσε πάντα να επαναφέρει ολόκληρο τον δίσκο σε προηγούμενή του κατάσταση, παρακάμπτοντας έτσι τον οποιοδήποτε έλεγχο. Ωστόσο, εάν είναι επιθυμητό κάποιο μη τελείως απόλυτο επίπεδο ακεραιότητας δίσκου, αυτό μπορεί να επιτευχθεί εντός του κρυπτογραφημένου δίσκου χρησιμοποιώντας κώδικες αυθεντικοποίησης μηνύματος (MAC, βλέπε Ενότητα 5.2) για κάθε αρχείο αποθήκευσης.

9.2.1.1 Κλασικοί Τρόποι Λειτουργίας και Βελτιώσεις

Όπως τα περισσότερα σχήματα κρυπτογράφησης, έτσι και η κρυπτογράφηση δίσκου βασίζεται σε κρυπτογράφηση μπλοκ χρησιμοποιώντας τρόπους λειτουργίας οι οποίοι επιτρέπουν την κρυπτογράφηση μεγαλύτερων ποσοτήτων δεδομένων από το μέγεθος μπλοκ του κρυπτοσυστήματος (που είναι συνήθως 128 bits). Επομένως, οι τρόποι λειτουργίας αποτελούν ουσιαστικά τους κανόνες με τους οποίους θα γίνει η επανειλημμένη εφαρμογή της κρυπτογράφησης μπλοκ.

Ο τρόπος λειτουργίας αλυσιδωτού τμήματος (Cipher Block Chaining – CBC) αποτελεί μια κοινή λειτουργία αλυσίδας κατά την οποία το κρυπτογραφημένο κείμενο του προηγούμενου μπλοκ αντιστοιχίζεται με το απλό κείμενο του τρέχοντος μπλοκ μέσω μιας πράξης XOR πριν από την κρυπτογράφηση:

$$C_i = E_K(C_{i-1} \oplus P_i)$$

Εφόσον το πρώτο μπλοκ της αλυσίδας (P_0) δεν διαθέτει προηγούμενο κρυπτογραφημένο μπλοκ, χρησιμοποιείται ένα Διάνυσμα Αρχικοποίησης (Initialisation Vector – IV) αντί του C_{-1} . Αυτό, με τη σειρά του, καθιστά την λειτουργία CBC προσαρμόσιμη κατά κάποιον τρόπο.

Ωστόσο, η λειτουργία CBC αντιμετωπίζει κάποια προβλήματα, για παράδειγμα, εάν τα IVs είναι προβλέψιμα, τότε ένας αντίταλος μπορεί να αφήσει ένα «υδατογράφημα» (watermark) στο δίσκο, δηλαδή να αποθηκεύσει ένα ειδικά δημιουργημένο αρχείο ή συνδυασμό αρχείων που είναι ταυτοποιήσιμα ακόμη και μετά την κρυπτογράφηση. Η ακριβής μέθοδος κατασκευής του υδατογραφήματος εξαρτάται από την επιμέρους συνάρτηση που παρέχει τα IVs, αλλά η γενική ιδέα είναι να δημιουργηθούν δύο κρυπτογραφημένοι τομείς (sectors) με πανομοιότυπα τα πρώτα μπλοκ b_1 και b_2 , και στη συνέχεια αυτά τα δύο σχετίζονται μεταξύ τους ως εξής: $b_1 \oplus IV_1 = b_2 \oplus IV_2$. Επομένως, η κρυπτογράφηση του b_1 είναι πανομοιότυπη με την κρυπτογράφηση του b_2 , αφήνοντας ένα είδος υδατογραφήματος στο δίσκο. Το ακριβές μοτίβο του «ιδιού-διαφορετικού-ιδιού-διαφορετικού» μπλοκ στο δίσκο μπορεί στη συνέχεια να τροποποιηθεί κατάλληλα ώστε το υδατογράφημα να γίνει μοναδικό για ένα δεδομένο αρχείο.

Για την προστασία από την επίθεση υδατογράφησης που περιγράφηκε προηγουμένως, χρησιμοποιείται ένα κρυπτοσύστημα ή μια συνάρτηση σύνοψης για τη δημιουργία των IVs με βάση το κλειδί και τον τρέχοντα αριθμό τομέα, έτσι ώστε ένας αντίταλος να μην μπορεί να προβλέψει τα IVs. Συγκεκριμένα, η προσέγγιση ESSIV [11] χρησιμοποιεί μια κρυπτογράφηση μπλοκ σε λειτουργία CTR για τη δημιουργία των IVs.

Το διάνυσμα αρχικοποίησης κρυπτογραφημένου αλατιού-τομέα (Encrypted Salt-Sector Initialization Vector – ESSIV) [11] είναι μια μέθοδος για τη δημιουργία διανυσμάτων αρχικοποίησης για κρυπτογράφηση μπλοκ με σκοπό την χρήση του στην κρυπτογράφηση δίσκου. Οι συνήθεις μέθοδοι για τη δημιουργία IVs είναι προβλέψιμες ακολουθίες αριθμών που βασίζονται, για παράδειγμα, σε μια χρονική σήμανση ή στον αριθμό τομέα, και επιτρέπουν ορισμένες επιθέσεις, όπως η επίθεση υδατογράφησης. Η προσέγγιση ESSIV αποτρέπει τέτοιους είδους επιθέσεις δημιουργώντας IVs, συνδυάζοντας τον αριθμό τομέα SN με την σύνοψη του κλειδιού

K, καθιστώντας με αυτό τον τρόπο το IV απρόβλεπτο:

$$IV(SN) = E_s(SN), \text{ όπου } s = \text{hash}(K)$$

Το ESSIV σχεδιάστηκε από τον Clemens Fruhwirth και έχει ενσωματωθεί στον πυρήνα Linux από την έκδοση 2.6.10, αν και ένα παρόμοιο σχήμα έχει χρησιμοποιηθεί για τη δημιουργία IVs για την κρυπτογράφηση της εικονικής μνήμης (swap) του OpenBSD από το 2000 [12]. Το ESSIV υποστηρίζεται προαιρετικά από συστήματα κρυπτογράφησης δίσκων, όπως το dm-crypt [13].

Ενώ η λειτουργία CBC (με ή χωρίς ESSIV) διασφαλίζει την εμπιστευτικότητα, δεν διασφαλίζει την ακεραιότητα των κρυπτογραφημένων δεδομένων. Εάν το απλό κείμενο (plaintext) είναι γνωστό στον επιτιθέμενο, είναι δυνατόν να αλλάξει κάθε δεύτερο μπλοκ απλού κειμένου με μια τιμή της επιλογής του, αλλάζοντας κατάλληλα τα ενδιάμεσα μπλοκ με τυχαίες τιμές. Κάτι τέτοιο μπορεί να χρησιμοποιηθεί για την πραγματοποίηση επιθέσεων στην κρυπτογράφηση δίσκου σε λειτουργία CBC ή CBC-ESSIV [14].

9.2.1.2 Ειδικές Προσεγγίσεις Τρόπων Λειτουργίας

Προκειμένου να αποφευχθούν τέτοιες είδους επιθέσεις, όπως αυτές που αναφέρθηκαν προηγουμένως, στους κλασικούς τρόπους λειτουργίας εισήχθησαν διαφορετικοί τρόποι λειτουργίας για την κρυπτογράφηση δίσκου, όπως οι LRW [15], XEX [16], XTS [17], CMC [18] και EME [19], οι οποίοι περιγράφονται παρακάτω.

Liskov, Rivest, και Wagner (LRW): Η λειτουργία LRW εισήχθη το 2002 από τους Liskov, Rivest και Wagner [15]. Αυτός ο τρόπος λειτουργίας χρησιμοποιεί δύο κλειδιά: το K είναι το κλειδί για την κρυπτογράφηση μπλοκ και το F είναι ένα πρόσθετο κλειδί ίδιου μεγέθους με το μπλοκ. Για παράδειγμα, για τον AES με κλειδί 256-bit, το K έχει μέγεθος 256-bit και το F έχει μέγεθος 128-bit, όσο και το μέγεθος του μπλοκ στον AES. Η κρυπτογράφηση του μπλοκ P με λογικό ευρετήριο I ενός δίσκου χρησιμοποιεί την ακόλουθη εξίσωση:

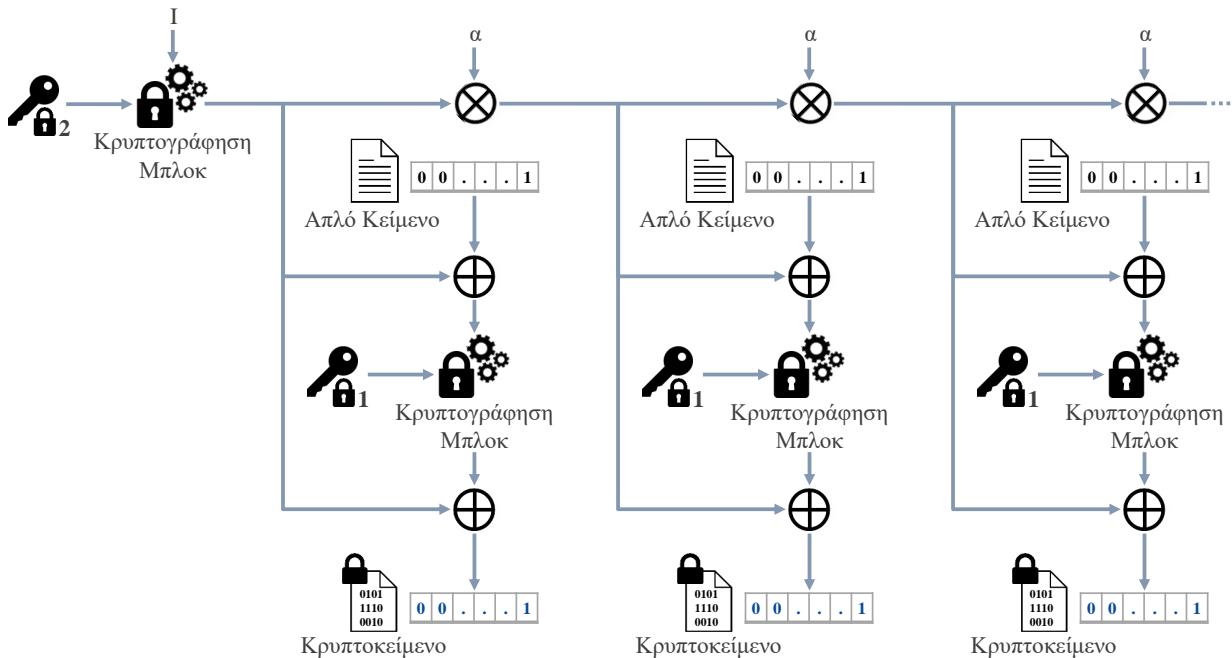
$$\begin{aligned} X &= F \otimes I \\ C &= E_K(P \oplus X) \oplus X \end{aligned}$$

Εδώ ο πολλαπλασιασμός \otimes και η πρόσθεση \oplus εκτελούνται σε πεπερασμένο πεδίο (π.χ. $\{0,1\}^{128}$ για τον AES). Με κάποιο προ-υπολογισμό, απαιτείται μόνο ένας πολλαπλασιασμός ανά τομέα (επίσης σημειώνεται ότι η πρόσθεση σε ένα δυαδικό πεπερασμένο πεδίο είναι η γνωστή πράξη XOR): $F \otimes I = F \otimes (I_0 \oplus \text{delta}) = F \otimes I_0 \oplus F \otimes \delta$, όπου οι $F \otimes \delta$ προ-υπολογίζονται εκ των προτέρων για όλες τις πιθανές τιμές του δ . Αυτός ο τρόπος λειτουργίας χρειάζεται μόνο μία κρυπτογράφηση ανά μπλοκ και προστατεύει από όλες τις επιθέσεις που αναφέρθηκαν προηγουμένως, εκτός από μια μικρή διαρροή: εάν ο χρήστης αλλάξει ένα μεμονωμένο μπλοκ απλού κειμένου σε έναν τομέα, τότε αλλάζει μόνο ένα μπλοκ κρυπτογραφημένου κειμένου. Η λειτουργία LRW χρησιμοποιείται από το BestCrypt [20] και υπάρχει επίσης ως προαιρετική επιλογή, για παράδειγμα, στο σύστημα κρυπτογράφησης δίσκων dm-crypt [13].

Xor-Encrypt-Xor (XEX): Η λειτουργία XEX σχεδιάστηκε από τον Rogaway [16] για να επιτρέπει την αποτελεσματική επεξεργασία διαδοχικών μπλοκ (ανάλογα με την κρυπτογράφηση που χρησιμοποιείται) εντός μιας μονάδας δεδομένων (π.χ. τομέας δίσκου). Η προσαρμογή της λειτουργίας XEX βασίζεται σε έναν συνδυασμό της διεύθυνσης τομέα και του ευρετήριον του μπλοκ εντός του τομέα. Το κρυπτοκείμενο C υπολογίζεται ως εξής:

$$\begin{aligned} X &= E_K(I) \otimes \alpha^j \\ C &= E_K(P \oplus X) \oplus X \end{aligned}$$

όπου το P είναι ένα απλό κείμενο, το I είναι ο αριθμός του τομέα, το α είναι ένα πρωταρχικό στοιχείο (primitive element) στο $\{0,1\}^{128}$ όπως ο αριθμός 2, και το j είναι ο αριθμός του μπλοκ εντός του τομέα. Στο Σχήμα 9.2 παρουσιάζεται και οπτικά ο τρόπος λειτουργίας XEX κατά την διαδικασία της κρυπτογράφησης, χωρίς να είναι απαραίτητη η χρήση δύο διαφορετικών κλειδιών, όπως φαίνεται και στο σχήμα 9.2.



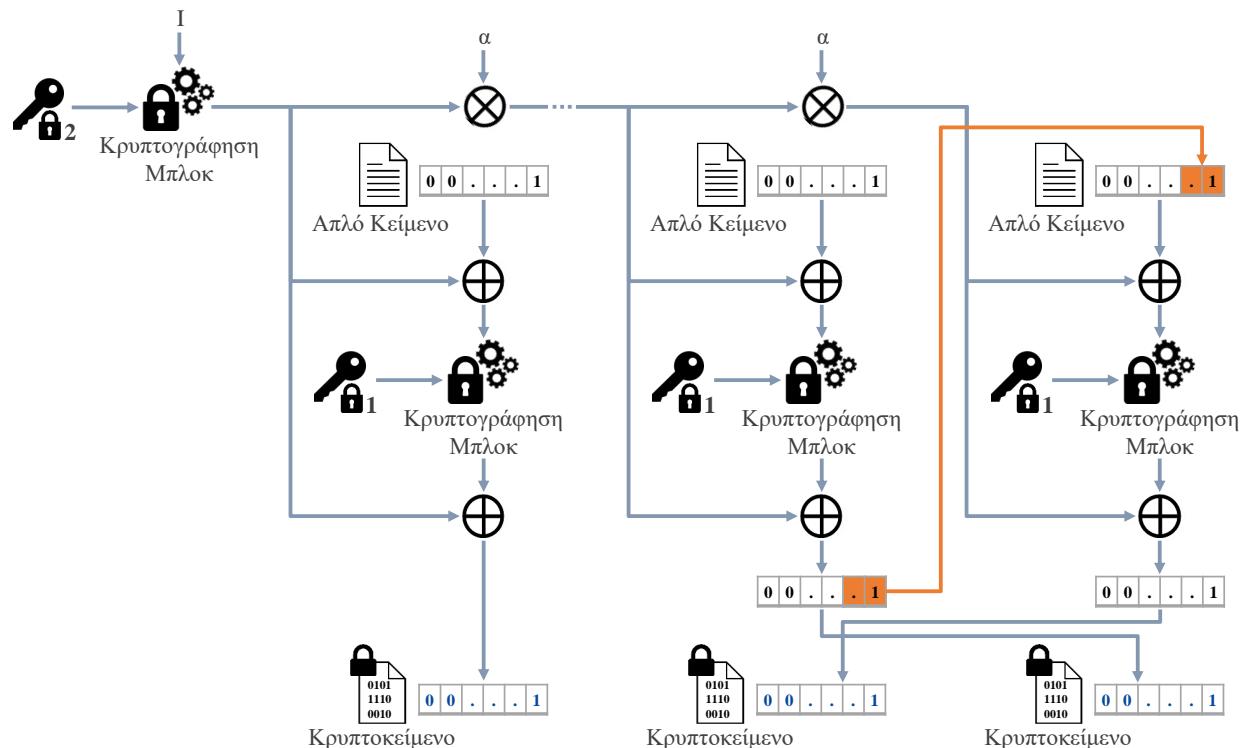
Σχήμα 9.2: Τρόπος λειτουργίας XEX κατά την κρυπτογράφηση.

XEX με χρήση κλοπής κρυπτοκειμένου (XTS): Η λειτουργία XEX σε συνδυασμό με την τεχνική κλοπής κρυπτοκειμένου (ciphertext stealing), γνωστή ως XTS, παρέχει υποστήριξη για τομείς δίσκου όπου το μέγεθός τους δεν διαιρείται ακριβώς με το μέγεθος μπλοκ, για παράδειγμα, τομείς με μέγεθος 520 bytes και μπλοκ των 16 bytes. Ο τρόπος με τον οποίο επιτυγχάνεται η λειτουργία XTS παρουσιάζεται στο Σχήμα 9.3. Το XTS-AES προτυποποιήθηκε τον Δεκέμβριο του 2007 ως το πρότυπο IEEE 1619-2007 [17]. Το πρότυπο αυτό υποστηρίζει τη χρήση διαφορετικού κλειδιού για την κρυπτογράφηση του IV από ότι για την κρυπτογράφηση του μπλοκ. Αυτό δείχνει να είναι αντίθετο με τον σκοπό του XEX και φαίνεται να πηγάζει από μια παρεμβινεία της αρχικής δημοσίευσης του XEX [16], αλλά φαίνεται ότι δεν έχει αρνητικές επιπτώσεις στην ασφάλειά του [21]. Αυτό έχει ως αποτέλεσμα, όταν γίνεται χρήση της κρυπτογράφησης AES-256 και AES-128 να πρέπει να παρέχονται κλειδιά των 512 bits και 256 bits, αντίστοιχα. Τον Ιανουαρίου του 2010, το NIST κυκλοφόρησε μία σύσταση με υπ' αριθμό SP 800-38E [22], η οποία ορίζει μία επιπλέον απαίτηση, η οποία περιορίζει το μέγιστο μέγεθος κάθε κρυπτογραφημένης μονάδας δεδομένων (συνήθως ενός τομέα ή μπλοκ δίσκου) σε 2^{20} μπλοκ AES. Σύμφωνα με το SP 800-38E, το XTS-AES παρέχει περισσότερη προστασία από τις άλλες εγκεκριμένες λειτουργίες εμπιστευτικότητας κατά της μη ξέουσιοδοτημένης τροποποίησης των κρυπτογραφημένων δεδομένων. Η λειτουργία XTS υποστηρίζεται από διάφορα συστήματα κρυπτογράφησης δίσκου, όπως το BestCrypt [20], το dm-crypt [13], το TrueCrypt [4], το VeraCrypt [23], και το BitLocker [24].

CBC–Mask–CBC (CMC): Η λειτουργία CMC, που εισήχθη από τους Halevi και Rogaway το 2003 [18], πραγματοποιεί τα εξής τρία βήματα: ολόκληρος ο τομέας του δίσκου είναι κρυπτογραφημένος σε λειτουργία CBC (έχοντας ως $C_{-1} = E_A(I)$), στην συνέχεια το κρυπτοκείμενο τροποποιείται κάνοντας την εξής πράξη $XOR\ 2(C'_0 \oplus C'_{k-1})$, και στο τέλος κρυπτογραφείται εκ νέου σε λειτουργία CBC ξεκινώντας από το τελευταίο μπλοκ. Το κύριο πρόβλημα με αυτή την λειτουργία είναι ότι για να αποκρυπτογραφήσει κανείς το P_0 πρέπει διαδοχικά να περάσει όλα τα δεδομένα δύο φορές.

ECB–Mask–ECB (EME): Για να λύσουν αυτό το πρόβλημα, οι Halevi και Rogaway εισήγαγαν το 2004 μια παραλληλοποιήσιμη παραλλαγή του CMC που ονομάζεται EME [19]. Η λειτουργία EME, με δεδομένο ότι τα απλά κείμενα είναι $P = P_1, \dots, P_m$ και τα κρυπτοκείμενα $C = C_1, \dots, C_m$, πραγματοποιεί τα εξής βήματα:

- Αρχικά, τα απλά κείμενα τροποποιούνται κάνοντας χρήση την πράξη XOR με το $L = E_K(0)$, μετατοπί-



Σχήμα 9.3: Τρόπος λειτουργίας XTS κατά την κρυπτογράφηση.

Ζονται κατά μια διαφορετική ποσότητα προς τα αριστερά, και κρυπτογραφούνται με βάση την εξίσωση: $P'_i = E_K(P_i \oplus 2^i L)$.

- Η μάσκα (mask) υπολογίζεται ως εξής: $M = M_P \oplus M_C$, όπου $M_P = I \oplus \bigoplus_{i=1}^{m-1} P'_i$ και $M_C = E_K(M_P)$.
- Τα ενδιάμεσα κρυπτοκείμενα τροποποιούνται με βάση την εξίσωση: $C'_i = P'_i \oplus 2^i M$ για $i = 1, \dots, m-1$ και $C'_0 = M_C \oplus I \oplus \bigoplus_{i=1}^{m-1} C'_i$.
- Τα τελικά κρυπτοκείμενα υπολογίζονται ως εξής: $C_i = E_K(C'_i) \oplus 2^i L$ για $i = 0, \dots, m-1$.

Στα παραπάνω βήματα ο συμβολισμός $\bigoplus_{i=1}^{m-1}$ υποδηλώνει διαδοχικές πράξεις XOR και αξίζει να σημειωθεί ότι σε αντίθεση με τις λειτουργίες LRW και CMC υπάρχει μόνο ένα κλειδί K .

9.2.2 BitLocker

Το BitLocker αποτελεί ένα σύστημα πλήρους κρυπτογράφησης δίσκου που περιλαμβάνεται στις εκδόσεις των Microsoft Windows από τα Windows Vista και μετά. Έχει σχεδιαστεί για την προστασία των δεδομένων παρέχοντας κρυπτογράφηση για ολόκληρα τμήματα (volume) δίσκου. Ως προεπιλογή, χρησιμοποιεί τον αλγόριθμο κρυπτογράφησης AES σε λειτουργία CBC ή λειτουργία XTS (στα Windows 10) κάνοντας χρήση ενός κλειδιού 128-bit ή 256-bit [24]. Θα πρέπει να επισημανθεί εδώ ότι η λειτουργία CBC δεν εφαρμόζεται σε ολόκληρο το δίσκο, αλλά μόνο σε κάθε επιμέρους τομέα (sector).

Τρεις είναι οι κύριοι μηχανισμοί αυθεντικοποίησης που μπορούν να χρησιμοποιηθούν ως δομικά στοιχεία για την εφαρμογή του BitLocker στην κρυπτογράφηση δίσκου:

- Διαφανής τρόπος λειτουργίας: Αυτή η λειτουργία χρησιμοποιεί τις δυνατότητες της μονάδας αξιόπιστης πλατφόρμας (TPM v1.2) για να παρέχει διαφανή εμπειρία κρυπτογράφησης στο χρήστη, χωρίς

να χρειάζεται να κάνει κάτι επιπλέον κατά την σύνδεσή του στα Windows. Το κλειδί που χρησιμοποιείται για την κρυπτογράφηση δίσκου είναι κρυπτογραφημένο από το TPM και θα παρασχεθεί στον κώδικα φόρτωσης του λειτουργικού συστήματος μόνο εάν τα αρχεία εκκίνησης δεν έχουν υποστεί κάποια αλλοίωση. Τα δομικά στοιχεία του BitLocker που θα εκτελεστούν πριν το λειτουργικό σύστημα, το επιτυγχάνουν αυτό εφαρμόζοντας μια μεθοδολογία που καθορίζεται από την Trusted Computing Group (TCG) [25]. Αυτή η λειτουργία είναι ευάλωτη σε μια επίθεση ψυχρής εκκίνησης (cold boot), καθώς επιτρέπει σε ένα μηχάνημα που έχει απενεργοποιηθεί να εκκινηθεί από έναν επιτιθέμενο και να διαβάσει το κλειδί που υπήρχε στην μνήμη RAM πριν απενεργοποιηθεί. Επίσης, είναι ευάλωτη σε μια επίθεση sniffing, καθώς το κλειδί κρυπτογράφησης του τμήματος του δίσκου μεταφέρεται σε απλό κείμενο από το TPM στον επεξεργαστή (CPU) κατά τη διάρκεια μιας επιτυχημένης εκκίνησης του μηχανήματος.

- **Λειτουργία αυθεντικοποίησης χρήστη:** Αυτή η λειτουργία απαιτεί από τον χρήστη να παρέχει κάποια στοιχεία για την αυθεντικοποίησή του στο περιβάλλον προ-εκκίνησης με τη μορφή PIN ή κάποιου κωδικού πρόσβασης, πριν την κανονική εκκίνηση του λειτουργικού συστήματος.
- **Λειτουργία χρήσης USB κλειδιού:** Σε αυτήν την λειτουργία, ο χρήστης πρέπει να εισάγει μια USB συσκευή που να περιέχει ένα κλειδί εκκίνησης στον υπολογιστή για να μπορέσει να εκκινηθεί το λειτουργικό σύστημα. Θα πρέπει να ληφθεί υπόψη ότι αυτή η λειτουργία απαιτεί το BIOS στο προστατευμένο μηχάνημα να υποστηρίζει την ανάγνωση συσκευών USB σε περιβάλλον εκκίνησης πριν από το λειτουργικό σύστημα. Το BitLocker αυτή την στιγμή δεν υποστηρίζει έξυπνες κάρτες (smart cards) για αυθεντικοποίηση κατά την προ-εκκίνηση [26].

Οι ακόλουθοι συνδυασμοί των παραπάνω μηχανισμών αυθεντικοποίησης είναι αυτοί που υποστηρίζονται και πάντα σε συνδυασμό με ένα προαιρετικό κλειδί ανάκτησης:

- Μόνο TPM
- TPM και PIN
- TPM, PIN και USB κλειδί
- TPM και USB κλειδί
- Μόνο USB κλειδί
- Μόνο κωδικός πρόσβασης

Όταν είναι ενεργοποιημένα, το TPM και το BitLocker, τότε μπορεί να διασφαλιστεί η ακεραιότητα της αξιόπιστης διαδρομής εκκίνησης (π.χ. το BIOS και ο τομέας εκκίνησης (MBR)), προκειμένου να αποτραπούν οι περισσότερες εκτός σύνδεσης φυσικές επιθέσεις και η εκτέλεση κακόβουλου λογισμικού στον τομέα εκκίνησης [26]. Για να κρυπτογραφήσει το BitLocker το τμήμα δίσκου που περιέχει το λειτουργικό σύστημα, απαιτούνται τουλάχιστον δύο τμήματα με μορφοποίηση NTFS [27]: ένα για το λειτουργικό σύστημα (συνήθως το C:) και άλλο ένα με ελάχιστο μέγεθος 100 MB, το οποίο παραμένει μη κρυπτογραφημένο και εκκινεί το λειτουργικό σύστημα.

Από την στιγμή που υπάρχει αυτό το εναλλακτικό τμήμα εκκίνησης, το TPM αρχικοποιείται (υποθέτοντας ότι υπάρχει αυτή η δυνατότητα) διαμορφώνοντας όλους τους απαιτούμενους μηχανισμούς προστασίας κλειδιού της κρυπτογράφησης δίσκου, όπως το TPM, η χρήση PIN ή USB κλειδιού (όπως αναφέρθηκε προηγουμένως). Στη συνέχεια, το τμήμα του λειτουργικού συστήματος κρυπτογραφείται ως μια εργασία παρασκενίου, κάτι που μπορεί να πάρει αρκετό χρόνο για έναν μεγάλο δίσκο, καθώς κάθε λογικός τομέας (sector) διαβάζεται, κρυπτογραφείται και ξαναγράφεται στο δίσκο. Τα κλειδιά προστατεύονται μόνο αφού κρυπτογραφηθεί

ολόκληρο το τμήμα και κατ' επέκταση όταν το τμήμα θεωρείται ασφαλές [28]. Επιπλέον, το BitLocker χρησιμοποιεί ένα πρόγραμμα οδήγησης συσκευής χαμηλού επιπέδου για την κρυπτογράφηση και την αποκρυπτογράφηση όλων των λειτουργιών στα αρχεία, καθιστώντας την αλληλεπίδραση με το κρυπτογραφημένο τμήμα διαφανή κατά την λειτουργία των εφαρμογών.

Το Σύστημα Κρυπτογράφησης Αρχείων (Encrypting File System – EFS) (περισσότερες λεπτομέρειες στην Ενότητα 9.3.1) μπορεί να χρησιμοποιηθεί σε συνδυασμό με το BitLocker για την παροχή προστασίας κατά την εκτέλεση του λειτουργικού συστήματος. Η προστασία των αρχείων από διεργασίες και χρήστες εντός του λειτουργικού συστήματος μπορεί να πραγματοποιηθεί μόνο με την χρήση λογισμικού κρυπτογράφησης που εκτελείται στα Windows, όπως το EFS που αναφέρθηκε. Επομένως, ο συνδυασμός του BitLocker και του EFS μπορούν να προσφέρουν προστασία σε μια ευρεία γκάμα διαφορετικών επιθέσεων.

9.3 Κρυπτογράφηση Δεδομένων σε Επίπεδο Συστήματος Αρχείων

Η κρυπτογράφηση δεδομένων σε επίπεδο συστήματος αρχείων, που συχνά ονομάζεται κρυπτογράφηση βάσει αρχείων (File-Based Encryption – FBE) ή κρυπτογράφηση αρχείων/φακέλων, είναι μια μορφή κρυπτογράφησης δίσκου όπου μεμονωμένα αρχεία ή ολόκληροι φάκελοι κρυπτογραφούνται από το ίδιο το σύστημα αρχείων (file system). Αυτό έρχεται σε αντιδιαστολή με την πλήρη κρυπτογράφηση δίσκου (FDE) όπου κρυπτογραφείται ολόκληρος ο δίσκος ή τμήμα (volume) του δίσκου, στον οποίο βρίσκεται το σύστημα αρχείων.

Στα πλεονεκτήματα της κρυπτογράφησης δεδομένων σε επίπεδο συστήματος αρχείων περιλαμβάνονται τα εξής:

- Παρέχεται μια ευέλικτη διαχείριση κλειδιών βάσει αρχείων, έτσι ώστε κάθε αρχείο να μπορεί (συνήθως) να κρυπτογραφείται με διαφορετικό κλειδί κρυπτογράφησης.
- Παρέχεται ατομική διαχείριση των κρυπτογραφημένων αρχείων, όπως για παράδειγμα, η δημιουργία σταδιακών αντιγράφων ασφαλείας των μεμονωμένων τροποποιημένων αρχείων ακόμη και σε κρυπτογραφημένη μορφή, αντί για την δημιουργία αντιγράφων ασφαλείας ολόκληρου του κρυπτογραφημένου τμήματος ενός δίσκου.
- Ο έλεγχος πρόσβασης μπορεί να επιβληθεί μέσω της χρήσης κρυπτογραφίας δημόσιου κλειδιού.
- Το γεγονός ότι τα κρυπτογραφικά κλειδιά διατηρούνται μόνο στη μνήμη, ενώ το αρχείο που αποκρυπτογραφείται από αυτά παραμένει ανοιχτό.

Δύο είναι οι κύριοι τύποι κρυπτογράφησης σε επίπεδο συστήματος αρχείων (περιγράφονται αναλυτικά παρακάτω):

- Η χρήση ενός κρυπτογραφικού συστήματος αρχείων (cryptographic file system) που τοποθετείται πάνω από το κύριο σύστημα αρχείων.
- Η χρήση ενός συστήματος αρχείων γενικού σκοπού (general-purpose file system) που υποστηρίζει κρυπτογράφηση.

Κρυπτογραφικά συστήματα αρχείων: Τα κρυπτογραφικά συστήματα αρχείων (cryptographic file system) [29] είναι έξειδικευμένα (όχι γενικού σκοπού) συστήματα αρχείων που έχουν σχεδιαστεί ειδικά με γνώμονα την κρυπτογράφηση και την ασφαλεία. Συνήθως κρυπτογραφούν όλα τα δεδομένα που περιέχουν, συμπεριλαμβανομένων των μεταδεδομένων, ωστόσο δεν θα πρέπει να συγχέονται με την πλήρη κρυπτογράφηση δίσκου. Αυτί να εφαρμόζουν μια μορφοποίηση στον δίσκο και τη δική τους κατανομή μπλοκ, αυτά τα συστήματα αρχείων συχνά τοποθετούνται πάνω από υπάρχοντα συστήματα αρχείων, για παράδειγμα, απλά βρίσκονται σε έναν φάκελο του υπάρχοντος συστήματος αρχείων. Πολλά τέτοια συστήματα αρχείων προσφέρουν επίσης προηγμένες δυνατότητες, όπως η αμφισβητήσιμη κρυπτογράφηση (deniable encryption) [30] (δηλ., η

ύπαρξη ενός κρυπτογραφημένου αρχείου είναι αμφισβητήσιμη με την έννοια ότι ένας αντίπαλος δεν μπορεί να αποδείξει ότι υπάρχει ή όχι), η επιβολή του δικαιώματος μόνο για ανάγνωση στο σύστημα αρχείων με κρυπτογραφικά ασφαλή τρόπο, καθώς και η διαφορετική προβολή της δομής ενός φακέλου ανάλογα με το κλειδί κρυπτογράφησης ή τον χρήστη.

Η χρήση ενός κρυπτογραφικού συστήματος αρχείων ενδείκνυται ειδικά σε περιπτώσεις όπου μέρος ενός υπάρχοντος συστήματος αρχείων συγχρονίζεται με το Νέφος. Σε τέτοιες περιπτώσεις, το κρυπτογραφικό σύστημα αρχείων θα μπορούσε να μπαίνει στο υψηλότερο επίπεδο και με αυτό τον τρόπο να συμβάλει στην προστασία του απορρήτου των δεδομένων. Παραδείγματα τέτοιων κρυπτογραφικών συστημάτων αρχείων είναι το eCryptfs [31] που ενσωματώνεται στον πυρήνα του Linux, και τα GocryptFS [32], CryFS [33] και SecureFS [34] που βασίζονται στο FUSE [35].

Συστήματα αρχείων γενικού σκοπού: Σε αντίθεση με τα κρυπτογραφικά συστήματα αρχείων ή την πλήρη κρυπτογράφηση δίσκου, τα συστήματα αρχείων γενικού σκοπού (general-purpose file system) που περιλαμβάνουν κρυπτογράφηση σε επίπεδο συστήματος αρχείων δεν κρυπτογραφούν συνήθως τα μεταδεδομένα του συστήματος αρχείων, όπως τη δομή των φακέλων, τα ονόματα αρχείων, το μέγεθος των αρχείων, και τις χρονικές σημάνσεις τροποποίησης. Ωστόσο, κάτι τέτοιο μπορεί να είναι προβληματικό εάν τα ίδια τα μεταδεδομένα πρέπει να διατηρηθούν εμπιστευτικά. Με άλλα λόγια, εάν τα αρχεία αποθηκεύονται με αναγνωριστικά ονόματα αρχείων, οποιοσδήποτε έχει πρόσβαση στον φυσικό δίσκο θα μπορούσε να γνωρίζει ποια έγγραφα είναι αποθηκευμένα στο δίσκο, χωρίς βέβαια να γνωρίζει το περιεχόμενό τους.

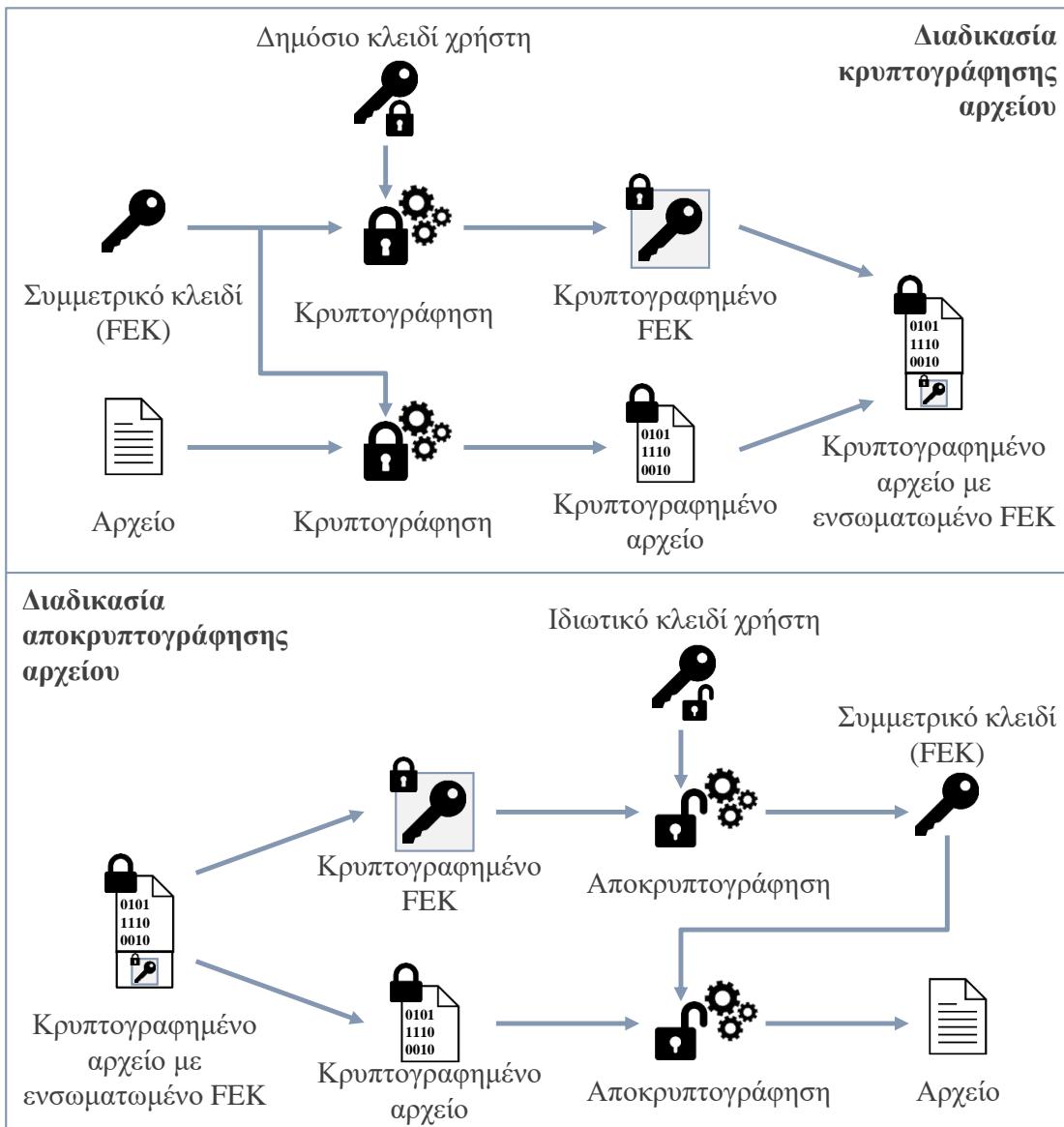
Μια εξαίρεση σε αυτόν τον κανόνα αποτελεί η υποστήριξη κρυπτογράφησης που υποστηρίζεται στο σύστημα αρχείων ZFS [10]. Τα μεταδεδομένα του συστήματος αρχείων, όπως τα ονόματα αρχείων, οι ιδιοκτήτες, οι λίστες ελέγχου πρόσβασης (Access-Control Lists – ACLs), και άλλα χαρακτηριστικά τους αποθηκεύονται όλα κρυπτογραφημένα στο δίσκο. Ωστόσο, τα μεταδεδομένα του ZFS που σχετίζονται με το χώρο αποθήκευσης, αποθηκεύονται σε μη κρυπτογραφημένη μορφή, οπότε είναι δυνατό να καθορίζονται τα διάφορα συστήματα αρχείων (σύνολα δεδομένων) που είναι διαθέσιμα στο χώρο αποθήκευσης, συμπεριλαμβανομένου και των κρυπτογραφημένων. Το περιεχόμενο των αποθηκευμένων αρχείων και φακέλων σε κάθε περίπτωση παραμένει κρυπτογραφημένο. Τέλος, άλλα γνωστά συστήματα αρχείων γενικού σκοπού είναι το NTFS (με χρήση του EFS) [27], το Ext4 [36], και το F2FS [37].

9.3.1 Σύστημα Κρυπτογράφησης Αρχείων (EFS)

Το Σύστημα Κρυπτογράφησης Αρχείων (Encrypting File System – EFS) στα Microsoft Windows είναι μια δυνατότητα που εισήχθη στην έκδοση 3.0 του NTFS [38] και παρέχει κρυπτογράφηση σε επίπεδο συστήματος αρχείων. Η τεχνολογία αυτή επιτρέπει στα αρχεία να κρυπτογραφούνται με διαφανή τρόπο για την προστασία των εμπιστευτικών δεδομένων από επιτιθέμενους που έχουν φυσική πρόσβαση σε έναν υπολογιστή. Το EFS είναι διαθέσιμο σε όλες τις εκδόσεις των Windows, εκτός από τις home εκδόσεις, από τα Windows 2000 και μετά. Από προεπιλογή, κανένα αρχείο δεν είναι κρυπτογραφημένο, αλλά η κρυπτογράφηση μπορεί να ενεργοποιηθεί από τον χρήστη ανά αρχείο, φάκελο ή μονάδα δίσκου. Ορισμένες από τις ρυθμίσεις του EFS μπορούν επίσης να απαιτηθούν μέσω πολιτικής ομάδας (group policy) σε περιβάλλοντα όπου υπάρχει ελεγκτής τομέα (domain controller) των Windows.

Το EFS λειτουργεί κρυπτογραφώντας ένα αρχείο με ένα συμμετρικό κλειδί (βλέπε Σχήμα 9.4), γνωστό και ως Κλειδί Κρυπτογράφησης Αρχείων (File Encryption Key – FEK). Χρησιμοποιεί έναν αλγόριθμο συμμετρικής κρυπτογράφησης επειδή χρειάζεται λιγότερος χρόνος για την κρυπτογράφηση και την αποκρυπτογράφηση μεγάλων ποσοτήτων δεδομένων από ότι εάν χρησιμοποιούταν μια κρυπτογράφηση δημοσίου κλειδιού. Ο αλγόριθμος συμμετρικής κρυπτογράφησης που χρησιμοποιείται ποικίλει (π.χ. AES, DESX και 3DES) ανάλογα με την έκδοση και τη διαμόρφωση του λειτουργικού συστήματος. Το FEK στη συνέχεια κρυπτογραφείται με ένα δημόσιο κλειδί που σχετίζεται με τον χρήστη που κρυπτογράφησε το αρχείο και αυτό το κρυπτογραφημένο FEK αποθηκεύεται στην εναλλακτική ροή δεδομένων \$EFS του κρυπτογραφημένου αρχείου. Για την αποκρυπτογράφηση του αρχείου, το πρόγραμμα οδήγησης του EFS χρησιμοποιεί το αντίστοιχο ιδιωτικό κλειδί για να αποκρυπτογράφησε το συμμετρικό κλειδί που είναι αποθηκευμένο στη ροή δεδομένων \$EFS.

Στη συνέχεια, το πρόγραμμα οδήγησης του EFS χρησιμοποιεί το συμμετρικό κλειδί για την αποκρυπτογράφηση των αρχείων. Επειδή οι λειτουργίες της κρυπτογράφησης και της αποκρυπτογράφησης εκτελούνται σε ένα επίπεδο κάτω από το NTFS, εκτελούνται με διαφανή τρόπο προς τον χρήστη και όλες του τις εφαρμογές.



Σχήμα 9.4: Λειτουργία του συστήματος κρυπτογράφησης αρχείων (EFS).

Οι φάκελοι των οποίων τα περιεχόμενα πρόκειται να κρυπτογραφηθούν από το EFS επισημάνονται με ένα χαρακτηριστικό κρυπτογράφησης. Το πρόγραμμα οδήγησης του EFS αντιμετωπίζει αυτό το χαρακτηριστικό κρυπτογράφησης με τρόπο ανάλογο με την κληρονομικότητα των δικαιωμάτων των αρχείων στο NTFS: εάν ένας φάκελος έχει επισημανθεί για κρυπτογράφηση, τότε από προεπιλογή όλα τα αρχεία και οι υποφάκελοι που δημιουργούνται κάτω από το φάκελο αυτό κρυπτογραφούνται επίσης. Όταν τα κρυπτογραφημένα αρχεία μετακινούνται σε ένα άλλο τμήμα (volume) δίσκου με μορφοποίηση NTFS, τα αρχεία παραμένουν κρυπτογραφημένα. Ωστόσο, υπάρχουν πολλές περιπτώσεις κατά τις οποίες το αρχείο θα μπορούσε να αποκρυπτογραφηθεί χωρίς ο χρήστης να το ξητήσει ρητά από τα Windows.

Τέλος, όταν τα αρχεία και οι φάκελοι πρόκειται να μετακινθούν ή αντιγραφούν σε έναν τμήμα δίσκου που έχει μορφοποιηθεί με άλλο σύστημα αρχείων, όπως το FAT32, τότε αυτά αποκρυπτογραφούνται πριν την διαδικασία αντιγραφής ή μετακίνησης. Τέλος, όταν τα κρυπτογραφημένα αρχεία αντιγράφονται μέσω δικτύου

χρησιμοποιώντας το πρωτόκολλο SMB/CIFS [39], τα αρχεία αποκρυπτογραφούνται πριν σταλούν μέσω του δικτύου.

9.4 Κρυπτογράφηση Δεδομένων σε Επίπεδο Βάσης Δεδομένων

Η κρυπτογράφηση δεδομένων σε επίπεδο βάσης δεδομένων [40] αναφέρεται στη χρήση τεχνικών κρυπτογράφησης για τη μετατροπή μιας βάσης δεδομένων απλού κειμένου σε μια (μερικώς) κρυπτογραφημένη βάση δεδομένων, καθιστώντας την έτσι μη αναγνώσιμη στον καθέναν, εκτός από εκείνους που έχουν στην κατοχή τους τα κλειδιά κρυπτογράφησης.

Για τη διατήρηση της εμπιστευτικότητας των δεδομένων, η επιβολή πολιτικών ελέγχου πρόσβασης που ορίζονται στο σύστημα διαχείρισης βάσεων δεδομένων (DataBase Management System – DBMS) είναι μια από τις πιο γνωστές μεθόδους. Μια πολιτική ελέγχου πρόσβασης, δηλαδή ένα σύνολο εξουσιοδοτήσεων, μπορεί να λάβει διαφορετικές μορφές ανάλογα με το υποκείμενο μοντέλο δεδομένων (π.χ. σχεσιακό, noSQL, XML, κτλ.) και τον τρόπο με τον οποίο χορηγούνται οι εξουσιοδοτήσεις, είτε μετά από διακριτικό έλεγχο πρόσβασης (Discretionary Access Control – DAC), έλεγχο πρόσβασης στη βάση ρόλων (Role Based Access Control – RBAC), ή υποχρεωτικό έλεγχο πρόσβασης (Mandatory Access Control – MAC).

Όποιο και αν είναι το μοντέλο ελέγχου πρόσβασης, οι εξουσιοδοτήσεις που επιβάλλονται από τον διακομιστή της βάσης δεδομένων μπορούν να παρακαμφθούν με διάφορους τρόπους. Για παράδειγμα, ένας εισβολέας μπορεί να διεισδύσει στο σύστημα πληροφοριών και να προσπαθήσει να διαβάσει το αρχείο της βάσης δεδομένων στο δίσκο. Μια άλλη πηγή απειλών προέρχεται από το γεγονός ότι πολλές βάσεις δεδομένων ανατίθενται σήμερα σε εξωτερικούς παρόχους υπηρεσιών βάσεων δεδομένων (Database Service Providers – DSPs). Αυτό έχει ως αποτέλεσμα, οι ιδιοκτήτες δεδομένων, μη έχοντας άλλη επιλογή, να εμπιστεύονται τους DSPs οι οποίοι απλά υποστηρίζουν ότι τα συστήματά τους είναι πλήρως ασφαλή και οι υπάλληλοι τους είναι πέρα από κάθε υποψία, μια υπόθεση που συχνά καταρρίπτεται από αντίθετα γεγονότα [41]. Τέλος, ένας διαχειριστής βάσης δεδομένων (DataBase Administrator – DBA) έχει ήδη αρκετά δικαιώματα για να παραβιάσει τον ορισμό του ελέγχου πρόσβασης και να κατασκοπεύσει τη συμπεριφορά του DBMS.

Έχοντας ως κανόνα μια παλιά και σημαντική αρχή που ονομάζεται άμυνα σε βάθος (δηλ., άμυνα σε πολλαπλά επίπεδα, έτσι ώστε οι επιτιθέμενοι να πρέπει να καταφέρουν να περάσουν όλα αυτά τα επίπεδα άμυνας), η χρήση κρυπτογραφικών τεχνικών για τη συμπλήρωση και την ενίσχυση του ελέγχου πρόσβασης μιας βάσης δεδομένων αποτελεί ένα από τα επιπλέον επίπεδα άμυνας [42]. Ο σκοπός της κρυπτογράφησης μιας βάσης δεδομένων είναι να διασφαλίσει ότι δεν θα αποκαλυφθούν τα περιεχόμενα μιας βάσης δεδομένων σε μη εξουσιοδοτημένα άτομα (π.χ. επιτιθέμενους). Ακόμα κι αν οι επιτιθέμενοι περάσουν από το τείχος προστασίας του συστήματος και παρακάμψουν τις πολιτικές ελέγχου πρόσβασης, εξακολουθούν να χρειάζονται τα κλειδιά κρυπτογράφησης για την αποκρυπτογράφηση των δεδομένων.

Η κρυπτογράφηση μπορεί να παρέχει ισχυρή ασφάλεια για δεδομένα σε κατάσταση ηρεμίας, αλλά η ανάπτυξη μιας στρατηγικής κρυπτογράφησης βάσεων δεδομένων πρέπει να λάβει υπόψη της διάφορους παραγοντες, όπως για παράδειγμα:

- Πού πρέπει να γίνει η κρυπτογράφηση: στο επίπεδο αποθήκευσης, στη βάση δεδομένων ή στην εφαρμογή όπου έχουν παραχθεί τα δεδομένα;
- Πόσα από τα δεδομένα πρέπει να είναι κρυπτογραφημένα για να παρέχεται επαρκής ασφάλεια;
- Ποιος πρέπει να είναι ο αλγόριθμος κρυπτογράφησης και ο τρόπος λειτουργίας;
- Ποιος πρέπει να έχει πρόσβαση στα κλειδιά κρυπτογράφησης;
- Πώς ελαχιστοποιείται ο αντίκτυπος της κρυπτογράφησης της βάσης δεδομένων σε σχέση με την απόδοση;

9.4.1 Επίπεδο Κρυπτογράφησης

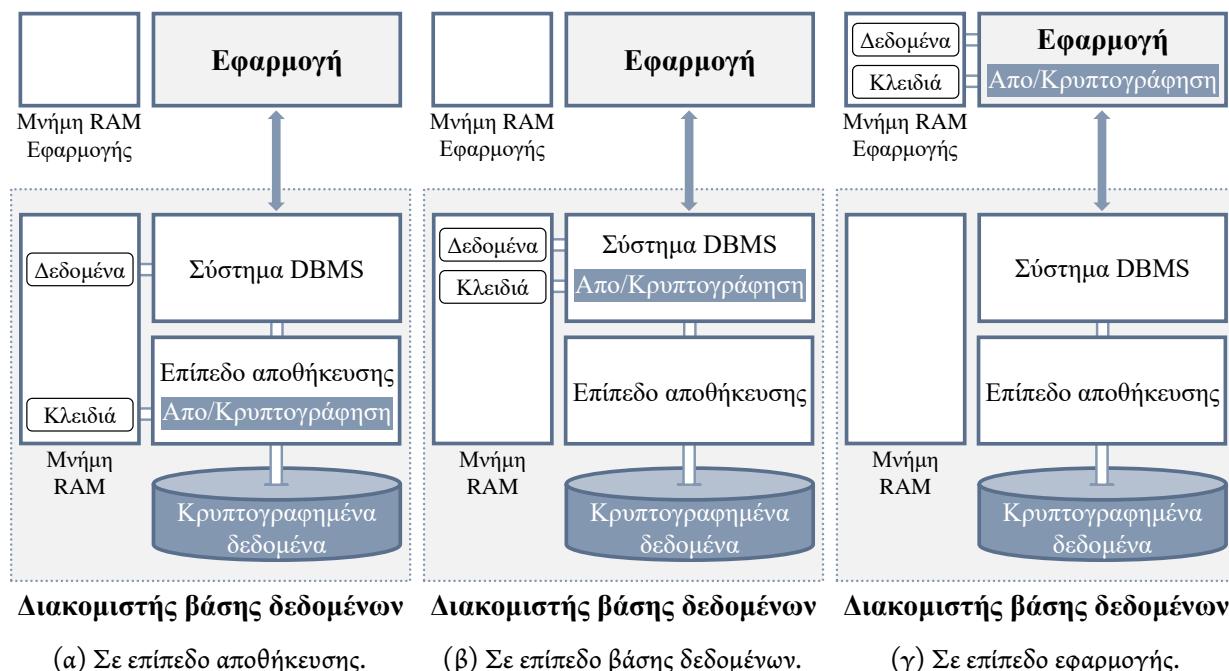
Κρυπτογράφηση σε επίπεδο αποθήκευσης: Αυτή η προσέγγιση ισοδυναμεί με την κρυπτογράφηση δεδομένων στο υποσύστημα αποθήκευσης (Σχήμα 9.5α) και έτσι προστατεύει τα δεδομένα σε κατάσταση ηρεμίας (π.χ. από κλοπή των μέσων αποθήκευσης). Είναι κατάλληλη για κρυπτογράφηση αρχείων ή ολόκληρων φακέλων στο περιβάλλον του λειτουργικού συστήματος (βλέπε Ενότητα 9.3). Από την πλευρά της βάσης δεδομένων, η κρυπτογράφηση σε επίπεδο αποθήκευσης έχει το πλεονέκτημα να είναι διαφανής, αποφεύγοντας έτσι τυχόν αλλαγές σε υπάρχουσες εφαρμογές. Από την άλλη πλευρά, καθώς το υποσύστημα αποθήκευσης δεν έχει γνώση των αντικειμένων και της δομής της βάσης δεδομένων, η στρατηγική κρυπτογράφησης δεν μπορεί να συσχετιστεί με τα δικαιώματα χρήστη (π.χ. χρήση διακριτών κλειδιών κρυπτογράφησης για διαφορετικούς χρήστες), και ούτε με ευαισθησία δεδομένων. Έτσι, η επιλεκτική κρυπτογράφηση – δηλαδή η κρυπτογράφηση μόνο τμημάτων της βάσης δεδομένων προκειμένου να μειωθεί η επιβάρυνση της κρυπτογράφησης – περιορίζεται σε επίπεδο αρχείου. Επιπλέον, η επιλεκτική κρυπτογράφηση αρχείων είναι επικίνδυνη, καθώς θα πρέπει να διασφαλιστεί ότι κανένα αντίγραφο ευαίσθητων δεδομένων δεν παραμένει μη κρυπτογραφημένο (π.χ. σε αρχεία καταγραφής, προσωρινά αρχεία, κ.λπ.).

Κρυπτογράφηση σε επίπεδο βάσης δεδομένων: Αυτή η τεχνική επιτρέπει την ασφάλεια των δεδομένων καθώς εισάγονται ή ανακτώνται από τη βάση δεδομένων (Σχήμα 9.5β). Η στρατηγική κρυπτογράφησης μπορεί επομένως να αποτελεί μέρος του σχεδιασμού της βάσης δεδομένων και μπορεί να σχετίζεται με την ευαισθησία των δεδομένων ή/και τα δικαιώματα χρήστη. Η επιλεκτική κρυπτογράφηση είναι δυνατή και μπορεί να γίνει σε διάφορα επίπεδα, όπως σε πίνακες, στήλες, και γραμμές. Μπορεί ακόμη και να συσχετιστεί με ορισμένες λογικές συνθήκες (π.χ. κρυπτογράφηση των μισθών που άνω των 10.000€/μήνα). Ανάλογα με το επίπεδο ενσωμάτωσης της δυνατότητας κρυπτογράφησης και του DBMS, η διαδικασία κρυπτογράφησης ενδέχεται να επιφέρει κάποιες αλλαγές στις εφαρμογές. Επιπλέον, μπορεί να προκαλέσει υποβάθμιση της απόδοσης του DBMS, καθώς η κρυπτογράφηση γενικά απαγορεύει τη χρήση ευρετηρίου (index) σε κρυπτογραφημένα δεδομένα. Εκτός και εάν χρησιμοποιούνται συγκεκριμένοι αλγόριθμοι κρυπτογράφησης ή τρόποι λειτουργίας (π.χ. κρυπτογράφηση διατήρησης σειράς, και λειτουργίας ECB που διατηρεί την ισότητα), οι οποίοι καθιστούν την ευρετηρίαση κρυπτογραφημένων δεδομένων εφικτή.

Και για τις δύο στρατηγικές κρυπτογράφησης που αναφέρθηκαν μέχρι στιγμής, τα δεδομένα αποκρυπτογραφούνται στον διακομιστή της βάσης δεδομένων κατά το χρόνο εκτέλεσης. Έτσι, τα κλειδιά κρυπτογράφησης πρέπει να μεταδίδονται ή να διατηρούνται μαζί με τα κρυπτογραφημένα δεδομένα στην πλευρά του διακομιστή, παρέχοντας έτσι περιορισμένη προστασία έναντι του διαχειριστή ή οποιουδήποτε εισβολέα που μπορεί να υποκλέψει την ταυτότητα του διαχειριστή. Επιπλέον, οι εισβολείς θα μπορούσαν να κατασκοπεύσουν τη μνήμη RAM και να ανακαλύψουν με αυτόν τον τρόπο τα κλειδιά κρυπτογράφησης ή τα μη κρυπτογραφημένα δεδομένα.

Κρυπτογράφηση σε επίπεδο εφαρμογής: Αυτή η στρατηγική μετακινεί τη διαδικασία κρυπτογράφησης και αποκρυπτογράφησης στις εφαρμογές που δημιουργούν τα δεδομένα (Σχήμα 9.5γ). Επομένως, η κρυπτογράφηση εκτελείται εντός της εφαρμογής που εισάγει τα δεδομένα στο σύστημα, τα δεδομένα αποστέλλονται κρυπτογραφημένα, και επομένως αποθηκεύονται και ανακτώνται κρυπτογραφημένα [43, 44], για να αποκρυπτογραφηθούν τελικά από την ίδια την εφαρμογή. Αυτή η προσέγγιση έχει το πλεονέκτημα να διαχωρίζει τα κλειδιά κρυπτογράφησης από τα κρυπτογραφημένα δεδομένα που είναι αποθηκευμένα στη βάση δεδομένων, καθώς τα κλειδιά δεν χρειάζεται ποτέ να φύγουν από την πλευρά της εφαρμογής. Ωστόσο, οι εφαρμογές πρέπει να τροποποιηθούν κατάλληλα ώστε να υιοθετήσουν αυτή τη λύση. Επιπλέον, ανάλογα με το επίπεδο κρυπτογράφησης, η εφαρμογή ενδέχεται να χρειαστεί να ανακτήσει ένα μεγαλύτερο σύνολο δεδομένων από αυτό που παραχωρήθηκε στον πραγματικό χρήστη, ανοίγοντας έτσι ζητήματα παραβίασης ασφαλείας. Επιπλέον, ο χρήστης (ή οποιοσδήποτε εισβολέας αποκτά πρόσβαση στο μηχάνημα όπου εκτελείται η εφαρμογή) μπορεί να παραβιάσει την εφαρμογή και να αποκτήσει πρόσβαση σε μη εξουσιοδοτημένα δεδομένα. Τέλος, μια τέτοια στρατηγική προκαλεί επιβαρύνσεις απόδοσης (το ευρετήριο για τα κρυπτογραφημένα δεδομένα δεν έχει χρήση) και απαγορεύει τη χρήση ορισμένων προηγμένων λειτουργιών της βάσης δεδομένων στα κρυ-

πτογραφημένα δεδομένα, όπως stored procedures (δηλ., κώδικας που είναι αποθηκευμένος στο DBMS που μπορεί να κοινοποιηθεί και να κληθεί από πολλές εφαρμογές) και triggers (δηλ., ο κώδικας πυροδοτείται όταν τροποποιηθούν ορισμένα δεδομένα στη βάση δεδομένων). Όσον αφορά την ευελιξία και τη διαχείριση κλειδιών, η κρυπτογράφηση σε επίπεδο εφαρμογής προσφέρει την υψηλότερη ευελιξία, καθώς το επίπεδο κρυπτογράφησης και τα κλειδιά κρυπτογράφησης μπορούν να επιλεγούν ανάλογα με τη εκάστοτε λογική της εφαρμογής.



Σχήμα 9.5: Τρεις επιλογές για το επίπεδο κρυπτογράφησης μιας βάσης δεδομένων.

9.4.2 Αλγόριθμοι Κρυπτογράφησης και Τρόποι Λειτουργίας

Ανεξάρτητα από τη στρατηγική κρυπτογράφησης, η ασφάλεια των κρυπτογραφημένων δεδομένων εξαρτάται επίσης από τον αλγόριθμο κρυπτογράφησης, το μέγεθος του κλειδιού κρυπτογράφησης και την προστασία που παρέχει. Ακόμη και αν έχει υιοθετηθεί ένας ισχυρός αλγόριθμος κρυπτογράφησης, όπως AES, το κρυπτογραφημένο κείμενο θα μπορούσε να αποκαλύψει πληροφορίες του απλού κειμένου εάν επιλεγεί ένας ακατάλληλος τρόπος λειτουργίας. Για παράδειγμα, εάν ο αλγόριθμος κρυπτογράφησης εφαρμόζεται σε λειτουργία ήλεκτρονικού βιβλίου κωδικών (ECB), τα ίδια μπλοκ απλού κειμένου κρυπτογραφούνται σε πανομιότυπα μπλοκ κρυπτοκειμένου, αποκαλύπτοντας έτσι επαναλαμβανόμενα μοτίβα. Στο γενικό πλαίσιο μιας βάσης δεδομένων, τα επαναλαμβανόμενα μοτίβα είναι σύνηθες φαινόμενο, καθώς πολλές εγγραφές θα μπορούσαν να έχουν τις ίδιες τιμές πεδίων, επομένως θα πρέπει να δίνεται μεγάλη προσοχή κατά την επιλογή του τρόπου λειτουργίας κρυπτογράφησης. Επιπλέον, απλές λύσεις που μπορεί να λειτουργούν σε άλλο πλαίσιο (π.χ. χρησιμοποιώντας τη λειτουργία CTR και ένα διάνυσμα αρχικοποίησης που βασίζεται στην διεύθυνση των δεδομένων) ενδέχεται να αποτύχουν στις βάσεις δεδομένων, καθώς τα δεδομένα μπορεί να αλλάξουν (σε σχέση με το προηγούμενο παράδειγμα, εκτελώντας την πράξη του XOR μεταξύ της παλιάς και της νέας έκδοσης των κρυπτογραφημένων δεδομένων θα αποκαλύψει το XOR μεταξύ της παλιάς και της νέας έκδοσης των αρχικών μη κρυπτογραφημένων δεδομένων). Θα πρέπει να λαμβάνονται υπόψη όλες οι ιδιαιτερότητες των βάσεων δεδομένων, όπως τα επαναλαμβανόμενα μοτίβα, οι αλλαγές δεδομένων, και ο τεράστιος όγκος κρυπτογραφημένων δεδομένων, για να κατευθυνθεί η σωστή επιλογή ενός επαρκούς αλγόριθμου κρυπτογράφησης και τρόπου λειτουργίας. Επιπλέον, η προστασία θα πρέπει να είναι αρκετά ισχυρή καθώς τα δεδομένα ενδέχεται να ισχύουν για πολύ μεγάλο χρονικό διάστημα (αρκετά χρόνια), και επομένως, θα πρέπει να χρη-

σιμοποιούνται υπερσύγχρονοι αλγόριθμοι κρυπτογράφησης και τρόποι λειτουργίας.

9.4.3 Διαχείριση Κλειδιών Κρυπτογράφησης

Η διαχείριση κλειδιών κρυπτογράφησης αναφέρεται στον τρόπο με τον οποίο γίνεται η παραγωγή και διαχείριση των κρυπτογραφικών κλειδιών καθ' όλη τη διάρκεια της ζωής τους. Επειδή η κρυπτογραφία βασίζεται σε κλειδιά τα οποία γίνεται η κρυπτογράφηση και η αποκρυπτογράφηση των δεδομένων, η προστασία της βάσης δεδομένων είναι τόσο καλή όσο η προστασία που παρέχουν τα κλειδιά αυτά. Η τοποθεσία στην οποία κρατούνται τα κλειδιά κρυπτογράφησης και οι περιορισμοί πρόσβασης σε αυτά είναι επομένως ιδιαίτερα σημαντική. Δεδομένου ότι το πρόβλημα είναι εντελώς ανεξάρτητο από το επίπεδο κρυπτογράφησης, το κείμενο που ακολουθεί αφορά μόνο την στρατηγική κρυπτογράφησης σε επίπεδο βάσης δεδομένων.

Για την κρυπτογράφηση σε επίπεδο βάσης δεδομένων, μια εύκολη λύση είναι να αποθηκεύονται τα κλειδιά σε έναν πίνακα ή αρχείο βάσης δεδομένων με περιορισμένη πρόσβαση, δυνητικά κρυπτογραφημένο από ένα κύριο (master) κλειδί (το οποίο είναι αποθηκευμένο κάπου στον διακομιστή της βάσης δεδομένων). Άλλα όλοι οι διαχειριστές με δικαιώματα πρόσβασης θα μπορούσαν επίσης να έχουν πρόσβαση σε αυτά τα κλειδιά και να αποκρυπτογραφήσουν τυχόν δεδομένα εντός του συστήματος χωρίς ποτέ να εντοπιστούν.

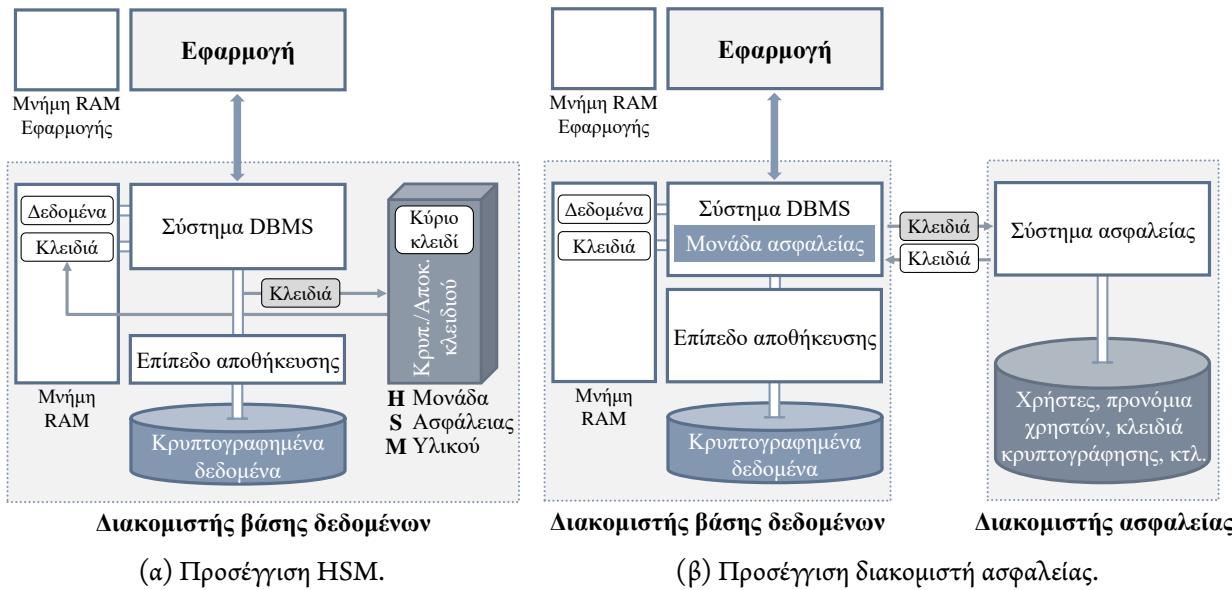
Για να ξεπεραστεί αυτό το πρόβλημα, μπορεί να χρησιμοποιηθεί εξειδικευμένο κρυπτογραφικό υλικό (π.χ. chipsets) ανθεκτικό σε παραβιάσεις, που ονομάζεται Μονάδα Ασφάλειας Υλικού (Hardware Security Module – HSM), για την παροχή ασφαλούς αποθήκευσης για τα κλειδιά κρυπτογράφησης [45, 46]. Γενικά, τα κλειδιά κρυπτογράφησης αποθηκεύονται στον διακομιστή κρυπτογραφημένα με χρήση ενός κυρίου κλειδιού που είναι αποθηκευμένο στο HSM. Κατά την χρονική στιγμή της κρυπτογράφησης και αποκρυπτογράφησης, τα κρυπτογραφημένα κλειδιά αποκρυπτογραφούνται δυναμικά από το HSM (χρησιμοποιώντας το κύριο κλειδί) και αφαιρούνται από τη μνήμη του διακομιστή αμέσως μόλις εκτελεστούν οι κρυπτογραφικές λειτουργίες, όπως φαίνεται στο Σχήμα 9.6a.

Μια εναλλακτική λύση είναι να μετακινηθούν εργασίες που σχετίζονται με την ασφάλεια σε ένα ξεχωριστό λογισμικό που εκτελείται σε έναν διαφορετικό διακομιστή, που ονομάζεται διακομιστής ασφαλείας, όπως φαίνεται στο Σχήμα 9.6β. Στη συνέχεια, ο διακομιστής ασφαλείας είναι αυτός που διαχειρίζεται τους χρήστες, τους κανόνες, τα προνόμια, τις πολιτικές κρυπτογράφησης και τα κλειδιά κρυπτογράφησης (ενδεχομένως να βασίζεται σε ένα HSM). Εντός του DBMS, μια μονάδα ασφαλείας επικοινωνεί με τον διακομιστή ασφαλείας για τον έλεγχο ταυτότητας των χρηστών, τον έλεγχο των δικαιωμάτων και την κρυπτογράφηση ή αποκρυπτογράφηση δεδομένων. Τα κλειδιά κρυπτογράφησης μπορούν επίσης να συνδεθούν με τον χρήστη ή με τα προνόμια του χρήστη. Επιπλέον, γίνεται σαφής διάκριση μεταξύ του ρόλου του DBA, που διαχειρίζεται τους πόρους της βάσης δεδομένων, και του ρόλου του διαχειριστή ασφαλείας (Security Administrator – SA), που διαχειρίζεται τις παραμέτρους ασφαλείας. Το όφελος της εμπιστοσύνης που παρέχει μια τέτοια λύση προέρχεται από το γεγονός ότι μια επίθεση απαιτεί να συνωμοτήσουν μεταξύ τους ο DBA και ο SA.

Ενώ η προσθήκη διακομιστή ασφαλείας ή/και HSM ελαχιστοποιεί την έκθεση των κλειδιών κρυπτογράφησης, δεν προστατεύει πλήρως τη βάση δεδομένων. Πράγματι, τα κλειδιά κρυπτογράφησης, καθώς και τα αποκρυπτογραφημένα δεδομένα εξακολουθούν να εμφανίζονται (έστω και για σύντομο χρονικό διάστημα) στη μνήμη RAM του διακομιστή της βάσης δεδομένων και μπορούν να γίνουν στόχος διαφόρων εισβολέων.

9.4.4 Εφαρμογές σε Συστήματα Διαχείρισης Βάσεων Δεδομένων

Εδώ και αρκετά χρόνια, οι περισσότεροι κατασκευαστές DBMS παρέχουν δυνατότητες κρυπτογράφησης που επιτρέπουν στους προγραμματιστές εφαρμογών να περιλαμβάνουν πρόσθετα μέτρα ασφάλειας δεδομένων μέσω επιλεκτικής κρυπτογράφησης των αποθηκευμένων δεδομένων. Τέτοιες δυνατότητες παρέχονται με τη μορφή εργαλείων ή πακέτων κρυπτογράφησης (Oracle 9i/10g [47]), συναρτήσεων που μπορούν να ενσωματωθούν σε SQL εντολές (IBM DB2 [48]) ή επεκτάσεων της γλώσσας SQL (Sybase/SAP [49] και SQL Server [50]). Για να περιοριστεί η επιβάρυνση στην απόδοση, η επιλεκτική κρυπτογράφηση μπορεί να γίνει γενικά σε επίπεδο στηλών ενός πίνακα, αλλά μπορεί ωστόσο να περιλαμβάνει αλλαγή του σχήματος της



Σχήμα 9.6: Διαφορετικές προσεγγίσεις διαχείρισης κρυπτογραφικών κλειδιών.

βάσης δεδομένων για να μπορέσει να φιλοξενήσει δυαδικά δεδομένα που προκύπτουν από τη διαδικασία κρυπτογράφησης [50].

Ο SQL Server 2008 εισάγει την διαφανή κρυπτογράφηση δεδομένων (Transparent Data Encryption – TDE) [45], η οποία είναι στην πραγματικότητα πολύ παρόμοια με την κρυπτογράφηση σε επίπεδο αποθήκευσης. Ολόκληρη η βάση δεδομένων προστατεύεται από ένα μόνο κλειδί (Database Encryption Key – DEK), το οποίο προστατεύεται από πιο σύνθετα μέσα, συμπεριλαμβανομένης της δυνατότητας χρήσης HSM. Το TDE εκτελεί όλες τις κρυπτογραφικές λειτουργίες σε επίπεδο εισόδου/εξόδου, αλλά εντός του συστήματος της βάσης δεδομένων, και δεν απαιτεί από τους προγραμματιστές εφαρμογών να προσαρμόσουν κατάλληλα τον κώδικα τους για την κρυπτογράφηση και την αποκρυπτογράφηση δεδομένων.

Η διαφανής κρυπτογράφηση δεδομένων (TDE) έχει εισαχθεί επίσης στην Oracle 10g/11g, διευρύνοντας σημαντικά τις δυνατότητες χρήσης κρυπτογραφίας στο DBMS [46]. Η διαχείριση των κλειδιών κρυπτογράφησης πραγματοποιείται από ένα HSM ή αποθηκεύονται σε ένα εξωτερικό αρχείο, γνωστό ως πορτοφόλι (wallet), το οποίο είναι κρυπτογραφημένο χρησιμοποιώντας έναν καθορισμένο κωδικό πρόσβασης από τον διαχειριστή. Η επιλεκτική κρυπτογράφηση μπορεί να γίνει σε επίπεδο στήλης ή σε υψηλότερο επίπεδο (π.χ. στο σύνολο αρχείων δεδομένων που αντιστοιχεί σε έναν ή περισσότερους πίνακες και ευρετήρια). Για να αποφευχθεί η ανάλυση των κρυπτογραφημένων δεδομένων, η Oracle προτείνει να συμπεριληφθεί στη διαδικασία κρυπτογράφησης κάποιο salt, δηλαδή, μια τυχαία συμβολοσειρά των 16 bytes που αποθηκεύεται μαζί με κάθε κρυπτογραφημένη τιμή. Ένα ενδιαφέρον, αλλά μάλλον επικίνδυνο χαρακτηριστικό είναι η δυνατότητα χρήσης της λειτουργίας κρυπτογράφησης που διατηρεί την ιδιότητα ελέγχου της ισότητας (συνήθως μια λειτουργία CBC με σταθερό διάνυσμα αρχικοποίησης), επιτρέποντας έτσι, για παράδειγμα, τη χρήση ευρετηρίων για την εύρεση των κρυπτογραφημένων τιμών που αναζητήθηκαν.

Η κρυπτογράφηση σε επίπεδο βάσης δεδομένων με την προσέγγιση χρήσης ενός διακομιστή ασφαλείας, που αναφέρθηκε στην προηγούμενη ενότητα, προτείνεται από την IBM DB2 με την χρήση του Data Encryption Expert (DEE) [48] καθώς και από άλλους τρίτους προμηθευτές, όπως η Protegrity [51], και τέτοιες λύσεις τρίτων μπορούν να προσαρμοστούν κατάλληλα στις περισσότερες βάσεις δεδομένων (Oracle, IBM DB2, SQL Server και Sybase/SAP).

9.4.5 Επιπτώσεις στην Απόδοση από την Χρήση Κρυπτογράφησης

Συνήθως, η κρυπτογράφηση στηλών θα πρέπει να περιορίζεται σε εκείνες τις στήλες που περιέχουν μόνο εναίσθητα δεδομένα ή προσωπικά αναγνωρίσιμες πληροφορίες (Personally Identifiable Information – PII),

όπως ο αριθμός κοινωνικής ασφάλισης, οι αριθμοί πιστωτικών καρτών, κτλ., προκειμένου να ελαχιστοποιήθούν οι επιπτώσεις της κρυπτογράφησης στην απόδοση. Επιπλέον, μπορεί να υπάρχει κάποια επιπλέον επιβάρυνση αποθήκευσης, ανάλογα με τη κρυπτογραφική στρατηγική που χρησιμοποιείται.

Σύμφωνα με την Oracle [46], η επίπτωση στην απόδοση είναι της τάξης του 4% έως 8% στον χρόνο απόκρισης του τελικού χρήστη καθώς και μια αύξηση στη χρήση του επεξεργαστή από 1% έως 5%. Στο DBMS Oracle 11.2.0.2 μπορεί να αξιοποιηθεί επιπλέον ο επιταχυντής κρυπτογράφησης AES-NI στον επεξεργαστή Intel Xeon 5600 (και μεταγενέστερων), γεγονός που καθιστά τις επιπτώσεις στην απόδοση ακόμη μικρότερες. Πριν από την χρήση κρυπτογράφησης σε επίπεδο βάσης δεδομένων, θα πρέπει να διερευνηθεί γενικότερα ο αντίκτυπος που θα έχει σε επίπεδο επεξεργαστή και μνήμης.

Επιπρόσθετα, η χρήση ευρετηρίου (index) ενδέχεται να είναι περιορισμένη σε ορισμένες μόνο λειτουργίες σύγκρισης (π.χ. ισότητας) στις κρυπτογραφημένες στήλες. Με αποτέλεσμα σε ορισμένες μάλιστα υλοποίησεις, που υποστηρίζεται απλά η ισοτιμία των κρυπτογραφημένων στηλών, να απαιτείται η πλήρης σάρωση ολόκληρου του πίνακα. Επίσης, δεν υποστηρίζονται όλοι οι τύποι δεδομένων για τις κρυπτογραφημένες στήλες, για παράδειγμα τα FILESTREAM δεδομένα στον SQL Server 2008 δεν υποστηρίζονται.

9.4.6 Κρυπτογράφηση των Αντιγράφων Ασφαλείας

Η κρυπτογράφηση των αντιγράφων ασφαλείας μιας βάσης δεδομένων είναι μια εξαιρετικά κρίσιμη διαδικασία για την ασφάλεια των (προσωπικών) δεδομένων. Μια από τις σημαντικότερες διαρροές προσωπικών δεδομένων που έχουν παρατηρηθεί στο παρελθόν [52] οφείλονται κατά κύριο λόγο στην ελλιπή προστασία των αντιγράφων ασφαλείας. Για το λόγο αυτό, θα πρέπει να δίνεται ιδιαίτερη προσοχή στην ασφάλεια των αντιγράφων ασφαλείας. Η ασφάλεια αυτή μπορεί να επιτευχθεί με την κρυπτογράφηση των δεδομένων με κάποιον συμμετρικό αλγόριθμο κρυπτογράφησης (π.χ. AES 256-bit) και τη χρήση κατάλληλου κλειδιού κρυπτογράφησης. Τα πλεονεκτήματα και μειονεκτήματα αυτής της κρυπτογράφησης των αντιγράφων ασφαλείας είναι τα εξής:

- Πλεονεκτήματα

- Πρώτα από όλα, η κρυπτογράφηση των αντιγράφων ασφαλείας σε έναν τοπικό σκληρό δίσκο μπορεί να αποτρέψει την πρόσβαση σε αυτά από μη εξουσιοδοτημένους χρήστες. Για παράδειγμα, εάν ο σκληρός δίσκος που περιέχει τα αντίγραφα ασφαλείας κλαπεί, δεν θα είναι δυνατή η πρόσβαση στα αντίγραφα ασφαλείας χωρίς την γνώση του κλειδιού κρυπτογράφησης.
- Επίσης, στην περίπτωση που τα αντίγραφα ασφαλείας αποθηκεύονται στο Νέφος (Cloud), η κρυπτογράφηση των αντιγράφων ασφαλείας είναι επίσης μια πολύ καλή λύση για την ενίσχυση της ασφάλειας των δεδομένων, αποτέλοντας με αυτό το τρόπο την πρόσβαση και την αλλοίωση αυτών από τους διάφορους cloud παρόχους.
- Τέλος, μέσω της κρυπτογράφησης των αντιγράφων ασφαλείας, παρέχεται μια παραπάνω σιγουριά ότι τα αντίγραφα ασφαλείας είναι ασφαλή, ανεξάρτητα από το μέσο αποθήκευσης και τους χρήστες που έχουν πρόσβαση σε αυτό.

- Μειονεκτήματα

- Χωρίς αμφιβολία, ένας από τους πιο συνηθισμένους κινδύνους της κρυπτογράφησης είναι η απώλεια του κλειδιού κρυπτογράφησης. Έτσι, εάν δεν εφαρμοστούν οι κατάλληλες στρατηγικές για την ασφαλή αποθήκευση των κλειδιών κρυπτογράφησης, η κρυπτογράφηση των αντιγράφων ασφαλείας μπορεί να αποβεί μοιραία χάνοντας την πρόσβαση σε κρίσιμα δεδομένα.
- Επιπλέον, εάν τα αντίγραφα ασφαλείας καταστραφούν για κάποιους λόγους (π.χ. bad sector σε σκληρό δίσκο), η κρυπτογράφηση δεδομένων θα κάνει την ανάκτηση των δεδομένων ακόμη πιο δύσκολη μιας και υπάρχει κίνδυνος να καταστραφούν περισσότερα δεδομένα σε σύγκριση με

το εάν τα δεδομένα δεν ήταν κρυπτογραφημένα. Ωστόσο, αυτό θα μπορούσε να ληφθεί υπόψη επιλέγοντας κάποιον καταλληλότερο αλγόριθμο κρυπτογράφησης.

Βιβλιογραφία

- [1] Ming Di Leom, Kim-Kwang Raymond Choo, and Ray Hunt. “Remote Wiping and Secure Deletion on Mobile Devices: A Review”. In: *Journal of Forensic Sciences* 61.6 (2016), pp. 1473–1492. doi: 10.1111/1556-4029.13203.
- [2] Erez Shmueli, Ronen Vaisenberg, Yuval Elovici, and Chanan Glezer. “Database Encryption: An Overview of Contemporary Challenges and Design Considerations”. In: *SIGMOD Record* 38.3 (Dec. 2010), pp. 29–34. issn: 0163-5808. doi: 10.1145/1815933.1815940.
- [3] Tim Maurer, Isabel Skierka, Robert Morgus, and Mirko Hohmann. “Technological Sovereignty: Missing the Point?” In: *7th International Conference on Cyber Conflict: Architectures in Cyberspace*. 2015, pp. 53–68. doi: 10.1109/CYCON.2015.7158468.
- [4] Alexei Czeskis et al. “Defeating Encrypted and Deniable File Systems: TrueCrypt v5.1a and the Case of the Tattling OS and Applications”. In: *Proceedings of the 3rd USENIX Workshop on Hot Topics in Security (HotSec '08)*. USENIX Association, 2008, pp. 1–7.
- [5] *Information technology. Trusted Platform Module Library - Commands*. Standard. London, UK: The British Standards Institution, Jan. 2016. doi: 10.3403/30302805U.
- [6] Michael O. Johnston and Stig Venaas. *Dynamic Host Configuration Protocol (DHCP) Options for the Intel Preboot eXecution Environment (PXE)*. RFC 4578. <https://www.rfc-editor.org/rfc/rfc4578.txt>. RFC Editor, Nov. 2006. doi: 10.17487/RFC4578.
- [7] J. Alex Halderman et al. “Lest We Remember: Cold-Boot Attacks on Encryption Keys”. In: *Communications of the ACM* 52.5 (May 2009), pp. 91–98. issn: 0001-0782. doi: 10.1145/1506409.1506429.
- [8] Karol Król and Dariusz Zdonek. “Peculiarity of the Bit Rot and Link Rot Phenomena”. In: *Global Knowledge, Memory and Communication* 69.1/2 (2019), pp. 20–37. doi: 10.1108/GKMC-06-2019-0067.
- [9] Ohad Rodeh, Josef Bacik, and Chris Mason. “BTRFS: The Linux B-Tree Filesystem”. In: *ACM Transactions on Storage* 9.3 (Aug. 2013). issn: 1553-3077. doi: 10.1145/2501620.2501623.
- [10] Jeff Bonwick, Matt Ahrens, Val Henson, Mark Maybee, and Mark Shellenbaum. “The Zettabyte File System”. In: *Proceedings of the 2nd USENIX Conference on File and Storage Technologies*. Vol. 215. USENIX Association, 2003, pp. 1–13.
- [11] Clemens Fruhwirth. *New Methods in Hard Disk Encryption*. <https://clemens.endorphin.org/nmihde/nmihde-A4-ds.pdf>. Institute for Computer Languages, Vienna University of Technology, June 2005.
- [12] Niels Provos. “Encrypting virtual memory”. In: *Proceedings of the 9th USENIX Security Symposium (USENIX Security '00)*. Denver, Colorado, USA: USENIX Association, 2000, pp. 35–44.
- [13] Milan Broz. *dm-crypt: Linux kernel Device-Mapper Crypto Target*. Wiki Page, accessed on 19 August 2022. <https://gitlab.com/cryptsetup/cryptsetup/wikis/DMCrypt>. 2015.
- [14] Jakob Lell. *Practical Malleability Attack Against CBC-Encrypted LUKS Partitions*. Blog Page, accessed on 19 August 2022. <http://www.jakoblell.com/blog/2013/12/22/practical-malleability-attack-against-cbc-encrypted-luks-partitions/>. 2013.

- [15] Moses Liskov, Ronald L. Rivest, and David Wagner. "Tweakable Block Ciphers". In: *Advances in Cryptology — CRYPTO 2002*. Ed. by Moti Yung. LNCS Vol. 2442. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 31–46. ISBN: 978-3-540-45708-4. DOI: 10.1007/3-540-45708-9_3.
- [16] Phillip Rogaway. "Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC". In: *Advances in Cryptology - ASIACRYPT 2004*. Ed. by Pil Joong Lee. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 16–31. ISBN: 978-3-540-30539-2. DOI: 10.1007/978-3-540-30539-2_2.
- [17] *IEEE Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices*. Standard. London, UK: Institute of Electrical and Electronics Engineers, Mar. 2008. DOI: 10.1109/IEEESTD.2008.4493450.
- [18] Shai Halevi and Phillip Rogaway. "A Tweakable Enciphering Mode". In: *Advances in Cryptology - CRYPTO 2003*. Ed. by Dan Boneh. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 482–499. ISBN: 978-3-540-45146-4. DOI: 10.1007/978-3-540-45146-4_28.
- [19] Shai Halevi and Phillip Rogaway. "A Parallelizable Enciphering Mode". In: *Topics in Cryptology – CT-RSA 2004*. Ed. by Tatsuaki Okamoto. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 292–304. ISBN: 978-3-540-24660-2. DOI: 10.1007/978-3-540-24660-2_23.
- [20] Mick Bauer. "Paranoid Penguin: BestCrypt: Cross-Platform Filesystem Encryption". In: *Linux Journal* 2002.98 (June 2002), p. 9. ISSN: 1075-3583.
- [21] Moses Liskov and Kazuhiko Minematsu. *Comments on XTS-AES*. Comments to NIST, accessed on 19 August 2022. http://csrc.nist.gov/groups/ST/toolkit/BCM/documents/comments/XTS/XTS_comments-Liskov_Minematsu.pdf. 2008.
- [22] Morris Dworkin. "Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices". In: *NIST Special Publication 800-38E* (2010), pp. 1–4. DOI: 10.6028/NIST.SP.800-38E.
- [23] VeraCrypt. *VeraCrypt - Free Open Source Disk Encryption with Strong Security for the Paranoid*. Official Website, accessed on 20 August 2022. <https://www.veracrypt.fr>. 2013.
- [24] Niels Ferguson. *AES-CBC + Elephant Diffuser: A Disk Encryption Algorithm for Windows Vista*. White Paper, accessed on 20 August 2022. <https://download.microsoft.com/download/0/2/3/0238acaf-d3bf-4a6d-b3d6-0a0be4bbb36e/bitlockercipher200608.pdf>. 2006.
- [25] Brian Berger. "Trusted Computing Group History". In: *Information Security Technical Report* 10.2 (2005), pp. 59–62. ISSN: 1363-4127. DOI: 10.1016/j.istr.2005.05.007.
- [26] Microsoft Corp. *Using BitLocker with Other Programs FAQ (Windows 10)*. Official Website, accessed on 21 August 2022. <https://docs.microsoft.com/en-us/windows/security/information-protection/bitlocker/bitlocker-using-with-other-programs-faq>. 2022.
- [27] Martin Karresand, Stefan Axelsson, and Geir Olav Dyrkolbotn. "Using NTFS Cluster Allocation Behavior to Find the Location of User Data". In: *Digital Investigation* 29 (2019), S51–S60. ISSN: 1742-2876. DOI: 10.1016/j.diin.2019.04.018.
- [28] Ed Bott. *Introducing Windows 10 for IT Professionals*. Microsoft Press, 2016. ISBN: 978-0-7356-9697-6.
- [29] Matt Blaze. "A Cryptographic File System for UNIX". In: *Proceedings of the 1st ACM Conference on Computer and Communications Security*. CCS '93. Fairfax, Virginia, USA: Association for Computing Machinery, 1993, pp. 9–16. ISBN: 0897916298. DOI: 10.1145/168588.168590.

- [30] Rein Canetti, Cynthia Dwork, Moni Naor, and Rafail Ostrovsky. "Deniable Encryption". In: *Advances in Cryptology — CRYPTO '97*. Ed. by Burton S. Kaliski. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 90–104. ISBN: 978-3-540-69528-8. DOI: 10.1007/BFb0052229.
- [31] Michael Austin Halcrow. "eCryptfs: An Enterprise-class Encrypted Filesystem for Linux". In: *Proceedings of the Linux Symposium*. Vol. 1. Ottawa, Ontario, Canada, 2005, pp. 201–218.
- [32] GocryptFS Project. *GocryptFS - An Encrypted Overlay Filesystem Written in Go*. GitHub Project, accessed on 21 August 2022. <https://github.com/rfjakob/gocryptfs>. 2016.
- [33] CryFS Project. *CryFS - A Cryptographic Filesystem for the Cloud*. GitHub Project, accessed on 21 August 2022. <https://github.com/cryfs/cryfs>. 2016.
- [34] SecureFS Project. *SecureFS - A Filesystem in Userspace (FUSE) with Transparent Authenticated Encryption*. GitHub Project, accessed on 21 August 2022. <https://github.com/nether196/securefs>. 2016.
- [35] FUSE Project. *FUSE - The Reference Implementation of the Linux FUSE (Filesystem in Userspace) Interface*. GitHub Project, accessed on 21 August 2022. <https://github.com/libfuse/libfuse>. 2015.
- [36] Avantika Mathur, Mingming Cao, and Andreas Dilger. "Ext4: The Next Generation of the Ext3 File System". In: *The USENIX Magazine* 32.3 (2007), pp. 25–30.
- [37] Changman Lee, Dongho Sim, Jooyoung Hwang, and Sangyeun Cho. "F2FS: A New File System for Flash Storage". In: *Proceedings of the 13th USENIX Conference on File and Storage Technologies (FAST '15)*. Santa Clara, CA, USA: USENIX Association, 2015, pp. 273–286.
- [38] Microsoft Corp. *File Encryption*. Official Website, accessed on 21 August 2022. <https://docs.microsoft.com/en-us/windows/win32/fileio/file-encryption>. 2021.
- [39] Samba Team. *Samba - Opening Windows to a Wider World*. Official Website, accessed on 21 August 2022. <https://www.samba.org>. 1992.
- [40] Luc Bouganim and Yanli Guo. "Database Encryption". In: *Encyclopedia of Cryptography and Security*. Ed. by Henk C. A. van Tilborg and Sushil Jajodia. Boston, MA: Springer US, 2011, pp. 307–312. ISBN: 978-1-4419-5906-5. DOI: 10.1007/978-1-4419-5906-5_677.
- [41] Hakan Hacigumus, Bala Iyer, and Sharad Mehrotra. "Providing database as a service". In: *Proceedings 18th International Conference on Data Engineering*. 2002, pp. 29–38. DOI: 10.1109/ICDE.2002.994695.
- [42] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. "Hippocratic Databases". In: *Proceedings of the 28th International Conference on Very Large Databases (VLDB '02)*. Ed. by Philip A. Bernstein, Yannis E. Ioannidis, Raghu Ramakrishnan, and Dimitris Papadias. San Francisco: Morgan Kaufmann, 2002, pp. 143–154. ISBN: 978-1-55860-869-6. DOI: 10.1016/B978-155860869-6/50021-4.
- [43] Luc Bouganim and Philippe Pucheral. "Chip-Secured Data Access: Confidential Data on Untrusted Servers". In: *Proceedings of the 28th International Conference on Very Large Databases (VLDB '02)*. Ed. by Philip A. Bernstein, Yannis E. Ioannidis, Raghu Ramakrishnan, and Dimitris Papadias. San Francisco: Morgan Kaufmann, 2002, pp. 131–142. ISBN: 978-1-55860-869-6. DOI: 10.1016/B978-155860869-6/50020-2.
- [44] Ernesto Damiani, S. De Capitani Vimercati, Sushil Jajodia, Stefano Paraboschi, and Pierangela Samarati. "Balancing Confidentiality and Efficiency in Untrusted Relational DBMSs". In: *Proceedings of the 10th ACM Conference on Computer and Communications Security*. CCS '03. Washington D.C., USA: Association for Computing Machinery, 2003, pp. 93–102. ISBN: 1581137389. DOI: 10.1145/948109.948124.

- [45] Michael Coles and Rodney Landrum. “Transparent Data Encryption”. In: *Expert SQL Server 2008 Encryption*. Berkeley, CA: Apress, 2009, pp. 127–150. ISBN: 978-1-4302-3365-7. doi: 10.1007/978-1-4302-3365-7_6.
- [46] Oracle Corp. *Oracle Advanced Security Transparent Data Encryption Best Practices*. White Paper, accessed on 23 March 2020. <http://www.oracle.com/technetwork/database/security/twp-transparent-data-encryption-bes-130696.pdf>. 2012.
- [47] Steven Feuerstein and Arup Nanda. “Data Encryption and Hashing”. In: *Oracle PL/SQL for DBAs*. 1st ed. O'Reilly Media, 2005. Chap. 4. ISBN: 9780596005870.
- [48] IBM Corp. *IBM Database Encryption Expert: Securing Data in DB2*. Technical White Paper, accessed on 24 August 2022. <ftp://ftp.software.ibm.com/software/data/db2imstools/whitepapers/IMW14003-USEN-01.pdf>. 2007.
- [49] SAP SE. *SAP Adaptive Server Enterprise: Database Encryption*. Technical Documentation, accessed on 24 August 2022. https://help.sap.com/docs/SAP_ASE. 2022.
- [50] Microsoft Corp. *SQL Server Encryption*. Official Website, accessed on 24 August 2022. <https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/sql-server-encryption>. 2021.
- [51] Ulf T. Mattsson. “Transparent Encryption and Separation of Duties for Enterprise Databases - A Solution for Field Level Privacy in Databases”. In: *Available at SSRN 571422* (2004). doi: 10.2139/ssrn.571422.
- [52] Robert E. Holtfreter and Adrian Harrington. “Data Breach Trends in the United States”. In: *Journal of Financial Crime* 22.2 (2015). doi: 10.1108/JFC-09-2013-0055.

ΚΕΦΑΛΑΙΟ 10

ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ ΚΑΤΑ ΤΗΝ ΜΕΤΑΦΟΡΑ

Περίληψη

Τα δεδομένα είναι ιδιαίτερα ευάλωτα όταν είναι σε κίνηση και η προστασία τους όταν βρίσκονται σε αυτήν την κατάσταση απαιτεί τη χρήση κατάλληλων πρωτοκόλλων και μηχανισμών. Βασική προϋπόθεση για την ασφαλή ανταλλαγή δεδομένων είναι η δημιουργία και χρήση ενός ασφαλούς καναλιού επικοινωνίας το οποίο θα διασφαλίσει, κατ' ελάχιστον, την εμπιστευτικότητα, την ακεραιότητα και την αυθεντικοποίηση των προς ανταλλαγή δεδομένων. Δεν υπάρχει μια ενιαία λύση για όλα τα είδη των επικοινωνιών που να καλύπτει τις ιδιαιτερότητες και απαιτήσεις τους αναφορικά με την προστασία των δεδομένων. Ως εκ τούτου, διάφορα πρωτόκολλα έχουν προταθεί και χρησιμοποιούνται εκτεταμένα στις σύγχρονες επικοινωνίες. Στο κεφάλαιο αυτό αναλύονται τα ιδιαίτερα χαρακτηριστικά που πρέπει να προσφέρει ένα ασφαλές κανάλι επικοινωνίας και πως η κρυπτογραφία μπορεί να καλύψει αυτές τις ανάγκες με τη χρήση πρωτοκόλλων τα οποία έχουν σχεδιαστεί να λειτουργούν στα διάφορα επίπεδα της καθιερωμένης στοίβας πρωτοκόλλων επικοινωνίας TCP/IP (Ενότητα 10.1). Επιπλέον αναλύονται ένα σύνολο τυποποιημένων πρωτοκόλλων που χρησιμοποιούνται σε διάφορα περιβάλλοντα καλύπτοντας τις εκάστοτε ανάγκες τους, όπως το TLS (Ενότητα 10.2), IPSec (Ενότητα 10.3) και SSH (Ενότητα 10.4).

Προαπαιτούμενη γνώση: Κατανόηση των βασικών εννοιών και αλγορίθμων της Κρυπτογραφίας που παρατίθενται στα εισαγωγικά κεφάλαια (Κεφάλαιο 1 έως 3) αυτού του βιβλίου.

10.1 Εισαγωγή

Τα δεδομένα σε κίνηση (data in motion) γνωστά και ως δεδομένα σε μεταφορά (data in transit), όπως αναφέρθηκε και στο προηγούμενο κεφάλαιο, αποτελεί μια από τις τρεις βασικές καταστάσεις των δεδομένων, κατά την οποία τα δεδομένα διακινούνται από ένα δικτυακό κόμβο προς έναν άλλο. Καθώς τα δεδομένα σε κίνηση ταξιδεύουν μέσα από δημόσια και ανασφαλή δίκτυα, αντιμετωπίζουν σοβαρές κυβερνοαπειλές που περιλαμβάνουν υποκλοπές (ακούσιες και σκόπιμες), παραποιήσεις και διαγραφές. Οι φορείς αυτών των απειλών εκμεταλλεύονται ευπάθειες στα πρωτόκολλα επικοινωνίας, αδύναμους μηχανισμούς κρυπτογράφησης ή ελέγχου

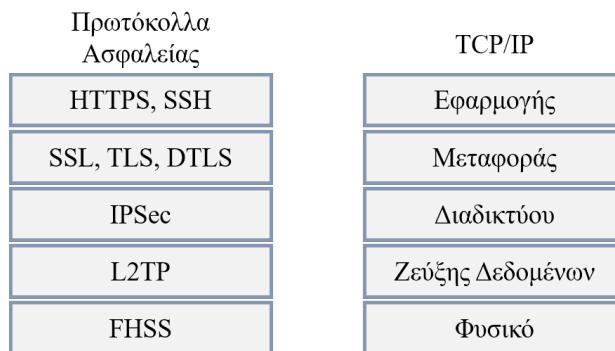
Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx-978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

ταυτότητας, ενώ μπορεί να αξιοποιούν και την πρόσβαση σε εσωτερικές πληροφορίες. Τα κακόβουλα ωφέλιμα φορτία (payload), ως αποτέλεσμα της ενεργούς παρακολούθησης των επικοινωνιών και της έγχυσης κακόβουλων δεδομένων σε αυτές, επιδεινώνουν περαιτέρω τις προκλήσεις ασφαλείας. Σε μια διαφορετική προσέγγιση, η ανάλυση της κυκλοφορίας επιτρέπει στους επιτιθέμενους να αντλήσουν πολύτιμες πληροφορίες από μοτίβα στη μετάδοση δεδομένων. Η αναγνώριση και η αντιμετώπιση αυτών των απειλών είναι απαραίτητη για τη διασφάλιση της εμπιστευτικότητας, της ακεραιότητας και της αυθεντικοποίησης των δεδομένων κατά τη διακίνησή τους, απαιτώντας μεταξύ άλλων, την εφαρμογή ισχυρής κρυπτογράφησης, ασφαλών πρωτοκόλλων επικοινωνίας, μηχανισμών ανίχνευσης παραποίησης των δεδομένων, αλλά και εκπαίδευσης των χρηστών για ενίσχυση της άμυνας έναντι των συνεχώς εξελισσόμενων κινδύνων.

Οι μηχανισμοί και τα πρωτόκολλα ασφαλείας που έχουν σχεδιαστεί για την προστασία των δεδομένων κατά τη μεταφορά, έχουν ως στόχο να εξασφαλίσουν, κατ' ελάχιστον, την μη εξουσιοδοτημένη πρόσβαση, τροποποίηση ή αποκάλυψη πληροφοριών. Δεν υπάρχει μοναδικά καθολική λύση για την προστασία των δεδομένων κατά τη μεταφορά καθώς, ανάλογα με το επίπεδο ασφάλειας που απαιτείται για τα εκάστοτε δεδομένα, το εύρος της προστασίας και τον αντίκτυπο στην απόδοση της επικοινωνίας, διαφορετικοί μηχανισμοί και πρωτόκολλα μπορούν να εφαρμοστούν σε διαφορετικά επίπεδα της αρχιτεκτονικής TCP/IP, για να εξασφαλίσουν το απαιτούμενο αποτέλεσμα.

Η αρχιτεκτονική TCP/IP αποτελείται από πέντε επίπεδα: το επίπεδο εφαρμογής (application layer), το επίπεδο μεταφοράς (transport layer), το επίπεδο δικτύου (network layer), το επίπεδο ζεύξης δεδομένων (data link layer) και το φυσικό επίπεδο (physical layer). Η προστασία των δεδομένων κατά τη μεταφορά μπορεί να παρέχεται σε κάποιο από αυτά τα επίπεδα, ως αποτέλεσμα της χρήσης ενός πρωτοκόλλου ασφαλείας το οποίο λειτουργεί στο αντίστοιχο TCP/IP επίπεδο, χωρίς να αποκλείεται η ταυτόχρονη χρήση πρωτοκόλλων σε διάφορα επίπεδα. Στο Σχήμα 10.1 απεικονίζονται ενδεικτικά πρωτόκολλα ασφαλείας στο TCP/IP επίπεδο το οποίο είναι σχεδιασμένα να παρέχουν προστασία, όπως είναι το HTTPS (Secure HTTP), SSH (Secure Shell), SSL (Secure Sockets Layers), TLS (Transport Layer Security), DTLS (Datagram TLS), IPSec (Internet Protocol Security), L2TP (Layer 2 Tunneling Protocol), FHSS (Frequency Hopping Spread Spectrum).



Σχήμα 10.1: Πρωτόκολλα ασφαλείας στην στοίβα TCP/IP.

Τα πρωτόκολλα ασφαλείας που θα εξετάσουμε σε αυτό το κεφάλαιο, αλλά και όλα όσα χρησιμοποιούνται γενικότερα, εφαρμόζουν συγκεκριμένους μηχανισμούς ασφαλείας, π.χ. κρυπτογράφηση ή αυθεντικοποίηση, παρέχοντας έτσι υπηρεσίες προστασίας των δεδομένων στα κατώτερα επίπεδα αλλά και κατά τη διακίνησή τους. Ωστόσο, καθώς κάθε επίπεδο έχει τα δικά του δυνατά και αδύναμα σημεία όσον αφορά τους μηχανισμούς ασφαλείας και τα πρωτόκολλα, μπορεί να χρειαστεί ένας συνδυασμός διαφορετικών πρωτοκόλλων σε διαφορετικά επίπεδα για να επιτευχθεί το βέλτιστο επίπεδο ασφάλειας για τα δεδομένα.

- Το επίπεδο εφαρμογής (application layer) είναι το υψηλότερο επίπεδο της αρχιτεκτονικής TCP/IP, όπου οι εφαρμογές και οι υπηρεσίες επικοινωνούν μεταξύ τους χρησιμοποιώντας διάφορα πρωτόκολλα. Παραδείγματα μηχανισμών και πρωτοκόλλων ασφαλείας επιπέδου εφαρμογής είναι τα HTTPS, SSH,

SFTP και SMTPS. Το κύριο χαρακτηριστικό όλων αυτών των μηχανισμών που λειτουργούν στο επίπεδο εφαρμογής είναι ότι μπορούν να παρέχουν από άκρο σε άκρο εμπιστευτικότητα, αυθεντικοποίηση και ακεραιότητα, εξασφαλίζοντας με αυτόν τον τρόπο την απαιτούμενη για πολλές εφαρμογές και υπηρεσίες, ασφάλεια από άκρο-σε-άκρο (end-to-end security).

Το πλεονέκτημα της εφαρμογής μηχανισμών ασφάλειας στο επίπεδο εφαρμογής είναι ότι μπορούν να προσφέρουν προσαρμοσμένη προστασία για διαφορετικούς τύπους δεδομένων και επικοινωνίας, εξυπηρετώντας έτσι καλύτερα τις ανάγκες της εκάστοτε εφαρμογής. Ένα βασικό μειονέκτημα ωστόσο είναι ότι, η κάθε εφαρμογή θα πρέπει να μεριμνήσει για την υλοποίηση των δικών της μηχανισμών, κατάλληλων για τα απαιτούμενα επίπεδα ασφάλειας.

- Το επίπεδο μεταφοράς (transport layer) είναι το επίπεδο κάτω από το επίπεδο εφαρμογής, όπου τα δεδομένα προετοιμάζονται κατάλληλα ώστε να μεταδοθούν χρησιμοποιώντας αξιόπιστα ή μη αξιόπιστα πρωτόκολλα. Παραδείγματα μηχανισμών και πρωτόκολλων ασφαλείας επιπέδου μεταφοράς είναι τα TLS, SSL και DTLS. Αυτοί οι μηχανισμοί και τα πρωτόκολλα παρέχουν εμπιστευτικότητα, αυθεντικοποίηση και ακεραιότητα σε ολόκληρη τη ροή δεδομένων του επιπέδου μεταφοράς.

Το πλεονέκτημα της προστασίας των δεδομένων στο επίπεδο μεταφοράς είναι ότι μπορεί να προσφέρει προστασία για οποιαδήποτε εφαρμογή ή υπηρεσία που χρησιμοποιεί τα πρωτόκολλα του επιπέδου μεταφοράς. Το μειονέκτημα είναι ότι ενδέχεται να μην μπορεί να προστατεύσει από ορισμένες επιθέσεις στο επίπεδο δικτύου, όπως η παραπλάνηση IP (IP spoofing).

- Το επίπεδο δικτύου φροντίζει για την δρομολόγηση των δεδομενογραμμάτων IP (IP datagrams) με τη χρήση λογικών διευθύνσεων. Το πιο χαρακτηριστικό παράδειγμα πρωτοκόλλου ασφαλείας στο επίπεδο δικτύου είναι το IPSec. Όπως και στα άλλα επίπεδα, οι μηχανισμοί και τα πρωτόκολλα ασφαλείας στο επίπεδο μεταφοράς τυπικά παρέχουν εμπιστευτικότητα σε ολόκληρο το ωφέλιμο φορτίο του δεδομενογράμματος, αλλά και αυθεντικοποίηση και ακεραιότητα για ολόκληρο το δεδομενόγραμμα του επιπέδου δικτύου.

Το πλεονέκτημα της εφαρμογής μηχανισμών ασφαλείας στο επίπεδο δικτύου είναι ότι μπορεί να προσφέρει ολοκληρωμένη και αποτελεσματική προστασία για οποιαδήποτε δεδομένα διασχίζουν το δίκτυο. Το μειονέκτημα είναι ότι μπορεί να απαιτεί περισσότερη διαμόρφωση και συντονισμό μεταξύ των συσκευών δικτύου, να μη παρέχει ασφάλεια από άκρο-σε-άκρο, εάν για παράδειγμα εφαρμόζεται μεταξύ δύο πυλών (gateways), και μπορεί να μην είναι σε θέση να προστατεύσει από ορισμένες επιθέσεις στο επίπεδο ζεύξης δεδομένων, όπως η πλαστογράφηση MAC (MAC Spoofing¹) ή η δηλητηρίαση ARP (ARP Poisoning²).

- Το επίπεδο ζεύξης δεδομένων (data link layer) είναι το χαμηλότερο επίπεδο της αρχιτεκτονικής TCP/IP, όπου τα δεδομενογράμματα IP μετατρέπονται σε πλαίσια και μεταδίδονται μέσω φυσικών μέσων. Παραδείγματα μηχανισμών και πρωτόκολλων ασφαλείας επιπέδου ζεύξης δεδομένων είναι τα PPTP, L2TP και 802.1X. Οι μηχανισμοί ασφαλείας του επιπέδου ζεύξης δεδομένων χρησιμοποιούνται για την προστασία από απειλές τοπικού δικτύου, όπως η υποκλοπή, η μη εξουσιοδοτημένη πρόσβαση και η παραβίαση δεδομένων στο ίδιο τμήμα δικτύου. Ωστόσο, δεν διασφαλίζουν την ασφάλεια των δεδομένων καθώς αυτά διασχίζουν πολλά διαφορετικά δίκτυα, συμπεριλαμβανομένου του Διαδικτύου, στο οποίο ενδέχεται να συμμετέχουν πολλοί ενδιάμεσοι κόμβοι.

¹Η πλαστογράφηση είναι μια τεχνική κλοπής ταυτότητας που χρησιμοποιείται από έναν επιτιθέμενο που υποδύεται μια αξιόπιστη οντότητα, σε αυτήν την περίπτωση μια άλλη συσκευή στο δίκτυο, παρουσιάζοντας μια διαφορετική φυσική διεύθυνση (MAC address) από αυτήν που πραγματικά έχει.

²Σε αυτή την επίθεση ο στόχος του επιτιθέμενου είναι να τροποποιήσει τις πληροφορίες στον πίνακα ARP (Address Resolution Protocol) όπου αποθηκεύεται προσωρινά η διεύθυνση IP που αντιστοιχεί με τη διεύθυνση MAC μιας συσκευής και να την αντικαταστήσει με τη δική του διεύθυνση MAC με σκοπό την ανακατεύθυνση των δεδομένων.

Το πλεονέκτημα της προστασίας των δεδομένων στο επίπεδο ζεύξης δεδομένων είναι ότι μπορεί να προσφέρει ισχυρή προστασία για οποιαδήποτε δεδομένα ταξιδεύουν πάνω από τα φυσικά μέσα. Το μειονέκτημα είναι ότι η προστασία αυτή τυπικά παρέχεται μεταξύ γειτονικών κόμβων αδυνατώντας να προσφέρει ασφάλεια από άκρο-σε-άκρο.

Στις ενότητες που ακολουθούν παρουσιάζονται σημαντικά πρωτόκολλα ασφαλείας, αντιπροσωπευτικά αυτών που συναντάμε στα προαναφερθέντα επίπεδα του TCP/IP.

10.2 Transport Layer Security

Το πρωτόκολλο Transport Layer Security (TLS) είναι ένα ευρέως διαδεδομένο πρωτόκολλο ασφαλείας που έχει σχεδιαστεί για να διευκολύνει την προστασία των δεδομένων στις επικοινωνίες μέσω ανασφαλών δικτύων, όπως είναι το Διαδίκτυο. Η κυριότερη ίσως χρήση του TLS είναι η κρυπτογράφηση της επικοινωνίας μεταξύ διαδικτυακών εφαρμογών και των χρηστών τους. Το TLS ωστόσο, μπορεί επίσης να χρησιμοποιηθεί για την κρυπτογράφηση και άλλων επικοινωνιών, όπως μηνυμάτων emails. Έτσι, χρησιμοποιείται για την ασφάλεια των επικοινωνιών σε μια μεγάλη ποικιλία διαδικτυακών συναλλαγών, όπως οικονομικές συναλλαγές (π.χ. τραπεζικές συναλλαγές, συναλλαγές μετοχών και ηλεκτρονικό εμπόριο) συναλλαγές υγειονομικής περίθαλψης (π.χ. προβολή ιατρικών αρχείων ή προγραμματισμός ιατρικών ραντεβού) και κοινωνικές συναλλαγές (π.χ. κοινωνική δικτύωση). Οποιαδήποτε υπηρεσία δικτύου που χειρίζεται εναίσθητα ή πολύτιμα δεδομένα, είτε πρόκειται για προσωπικά δεδομένα, οικονομικά δεδομένα ή διαπιστευτήρια, πρέπει να προστατεύει επαρκώς αυτά τα δεδομένα. Για όλες τις παραπάνω περιπτώσεις αλλά και πολλές άλλες, το TLS παρέχει ένα ασφαλές κανάλι για την αποστολή δεδομένων μεταξύ διακομιστή και πελάτη. Ο πελάτης είναι συχνά, αλλά όχι πάντα, ένα πρόγραμμα περιήγησης ιστού.

Το TLS είναι ένα πολυεπίπεδο πρωτόκολλο που λειτουργεί πάνω από ένα πρωτόκολλο αξιόπιστης μετάδοσης δεδομένων – συνήθως το Πρωτόκολλο Ελέγχου Μετάδοσης (Transmission Control Protocol – TCP). Τα πρωτόκολλα εφαρμογών, όπως το Hypertext Transfer Protocol (HTTP) και το Internet Message Access Protocol (IMAP), μπορούν να εκτελούνται πάνω από το TLS. Είναι ανεξάρτητο από εφαρμογές και προστατεύει τα δεδομένα αυτών χρησιμοποιώντας ένα σύνολο κρυπτογραφικών αλγορίθμων και μηχανισμών για τη διασφάλιση της εμπιστευτικότητας, της ακεραιότητας και της αυθεντικότητας των δεδομένων της εφαρμογής που ανταλλάσσονται.

10.2.1 Ιστορική Αναδρομή

Το πρωτόκολλο Secure Sockets Layer (SSL), το οποίο είναι ο πρόγονος του TLS, σχεδιάστηκε από την Netscape Corporation για να καλύψει τις ανάγκες ασφαλείας εφαρμογών πελάτη-διακομιστή. Η έκδοση 1 του SSL δεν κυκλοφόρησε ποτέ. Το SSL 2.0 κυκλοφόρησε το 1995 αλλά είχε πολλές γνωστές ευπάθειες, οι οποίες αντιμετωπίστηκαν με την κυκλοφορία του SSL 3.0 το 1996. Η Ομάδα Εργασίας Μηχανικής Διαδικτύου (Internet Engineering Task Force – IETF), μια τεχνική ομάδα εργασίας που είναι υπεύθυνη για την ανάπτυξη προτύπων Διαδικτύου για τη διασφάλιση της συμβατότητας των επικοινωνιών μεταξύ διαφορετικών εφαρμογών, προσπάθησε να επιλύσει όσο καλύτερα μπορούσε ζητήματα ασφαλείας και ασυμβατότητας μεταξύ των πρωτοκόλλων.

Η βελτίωση του SSL πρωτοκόλλου ήρθε με την έκδοση 1.0 του TLS, το οποίο τυποποιήθηκε από το IETF ως RFC 2246 [1]. Ενώ το TLS 1.0 βασίζεται στο SSL 3.0 και οι διαφορές μεταξύ τους δεν είναι πολλές, είναι αρκετά σημαντικές ώστε το TLS 1.0 και το SSL 3.0 να μη διαλείτουργούν.

Το TLS 1.1, (RFC 4346 [2]), αναπτύχθηκε για να αντιμετωπίσει τις αδυναμίες που ανακαλύφθηκαν στο TLS 1.0, κυρίως στα θέματα της επιλογής διανύσματος αρχικοποίησης για τον αλγόριθμο τύπου μπλοκ, και της επεξεργασίας σφαλμάτων συμπλήρωσης. Το TLS 1.2 (RFC 5246 [3]), έκανε αρκετές βελτιώσεις, ιδιαίτερα στον τομέα των συναρτήσεων σύνοψης, εισάγοντας τη δυνατότητα χρήσης της οικογένειας αλγορίθμων SHA-2, MAC και ψευδοτυχαίων αριθμών. Το TLS 1.2 προσθέτει επίσης την αυθεντικοποιημένη κρυπτογρά-

φηση με συσχετισμένα δεδομένα (Authenticated Encryption with Associated Data – AEAD).

Το TLS 1.3 (RFC 8446 [4]) εισήγαγε αρκετές αλλαγές, μεταξύ των οποίων ένα νέο πρωτόκολλο χειραψίας, μια νέα διαδικασία δημιουργίας κλειδιού που χρησιμοποιεί τη συνάρτηση Extract-and-Expand Key Derivation Function (HKDF) που βασίζεται στο HMAC, και την αφαίρεση αλγορίθμων κρυπτογράφησης που χρησιμοποιούν μεταφορά κλειδιού με RSA ή στατικό Diffie-Hellman, τον τρόπο λειτουργίας CBC καθώς και τον SHA-1. Πολλές επεκτάσεις που έχουν οριστεί για χρήση με το TLS 1.2 και τις προηγούμενες εκδόσεις δεν μπορούν να χρησιμοποιηθούν με το TLS 1.3.

10.2.2 Πρωτόκολλο TLS 1.3

Το πρωτόκολλο TLS αποτελείται από δύο επίπεδα, το πρωτόκολλο χειραψίας TLS (TLS Handshake Protocol) και το πρωτόκολλο εγγραφής TLS (TLS Record Protocol).

10.2.2.1 Πρωτόκολλο Χειραψίας TLS

Το Πρωτόκολλο Χειραψίας TLS (TLS Handshake Protocol) λειτουργεί πριν από το Πρωτόκολλο Εγγραφής TLS (βλέπε Ενότητα 10.2.2.2), προκειμένου να επιτρέψει στον διακομιστή και στον πελάτη να αυθεντικοποιήσουν ο ένας τον άλλον και να διαπραγματευτούν παραμέτρους ασφαλείας (αλγορίθμους κρυπτογράφησης και κρυπτογραφικά κλειδιά), πριν από τη μετάδοση δεδομένων. Το πρωτόκολλο χειραψίας έχει σχεδιαστεί για να αντιστέκεται σε παραβιάσεις. Ένας ενεργός εισβολέας δεν θα πρέπει να είναι σε θέση να αναγκάσει τις δύο πλευρές να διαπραγματευτούν διαφορετικές παραμέτρους από αυτές που πραγματικά επιθυμούν. Συγκεκριμένα, το Πρωτόκολλο Χειραψίας TLS έχει τις ακόλουθες ιδιότητες:

- Η ταυτότητα του διακομιστή και, προαιρετικά, του πελάτη επαληθεύονται χρησιμοποιώντας κρυπτογραφία δημοσίου κλειδιού.
- Οι δύο πλευρές διαπραγματεύονται με ασφάλεια τα κοινόχρηστα συμμετρικά κλειδιά τα οποία είναι διαθέσιμα μόνο σε αυτούς.
- Η διαπραγμάτευση είναι μια ασφαλής διαδικασία. Επιτιθέμενοι δεν μπορούν να αλλοιώσουν τη διαδικασία διαπραγμάτευσης χωρίς αυτό να εντοπιστεί από τις δύο πλευρές.

Επομένως, με το Πρωτόκολλο Χειραψίας, όταν ένας πελάτης και ένας διακομιστής TLS αρχίζουν να επικοινωνούν για πρώτη φορά, συμφωνούν σε μια έκδοση πρωτοκόλλου, επιλέγουν κρυπτογραφικούς αλγόριθμους, προαιρετικά ελέγχουν ο ένας την ταυτότητα του άλλου και χρησιμοποιούν μηχανισμούς κρυπτογραφίας δημοσίου κλειδιού για τη δημιουργία κοινών μυστικών κρυπτογραφικών κλειδιών συνόδου.

Το πρωτόκολλο χειραψίας TLS συνοπτικά περιλαμβάνει τα ακόλουθα βήματα:

- Ανταλλαγή μηνυμάτων “hello” που επιτρέπουν στον πελάτη και τον διακομιστή να διαπραγματευτούν τις παραμέτρους ασφάλειας, όπως την έκδοση του πρωτοκόλλου, τους κρυπτογραφικούς αλγόριθμους (συνίτες κρυπτογράφησης) και τις επεκτάσεις που θα χρησιμοποιηθούν, και να ανταλλάξουν κάποιες τυχαίες τιμές.
- Ανταλλαγή των απαραίτητων κρυπτογραφικών παραμέτρων ώστε πελάτης και διακομιστής να συμφωνήσουν σε ένα προ-κύριο (pre-master) μυστικό κλειδί.
- Ανταλλαγή πιστοποιητικών για να επιτραπεί στον πελάτη και τον διακομιστή να αποδείξουν τις ταυτότητες τους.
- Δημιουργία κύριων μυστικών κλειδιών από το προ-κύριο και τις τυχαίες τιμές που ανταλλάσσονται.

Το Σχήμα 10.2 απεικονίζει τα μηνύματα που ανταλλάσσονται σε μια πλήρη χειραψία στο TLS 1.3 και τα οποία είναι τα ακόλουθα:

1. Ανταλλαγή κλειδιών (αναλύεται περαιτέρω στην Ενότητα 10.2.2.1.1):

- Αποστολή ClientHello μηνύματος από τον πελάτη στον διακομιστή: Ο πελάτης ξεκινά τη χειραψία στέλνοντας ένα μήνυμα “hello” στον διακομιστή. Το μήνυμα περιλαμβάνει την έκδοση TLS και τις σούντες κρυπτογράφησης που υποστηρίζει ο πελάτης, καθώς και έναν τυχαίο αριθμό που είναι γνωστός ως “client random”.
- Ο διακομιστής επεξεργάζεται το μήνυμα ClientHello και καθορίζει τις κατάλληλες κρυπτογραφικές παραμέτρους για τη σύνδεση. Στη συνέχεια, αποκρίνεται με το δικό του μήνυμα ServerHello, το οποίο περιέχει τις κρυπτογραφικές παραμέτρους σύνδεσης ως αποτέλεσμα διαπραγμάτευσης με τον πελάτη. Για το TLS 1.3, το μήνυμα ServerHello καθορίζει τις επιλογές μόνο για το κρυπτογραφικό κλειδί και τον αλγόριθμο κρυπτογράφησης. Με το ServerHello ο διακομιστής στέλνει και αυτός στον πελάτη έναν τυχαίο αριθμό “server random”, που δημιουργεί για αυτή τη σύνοδο.

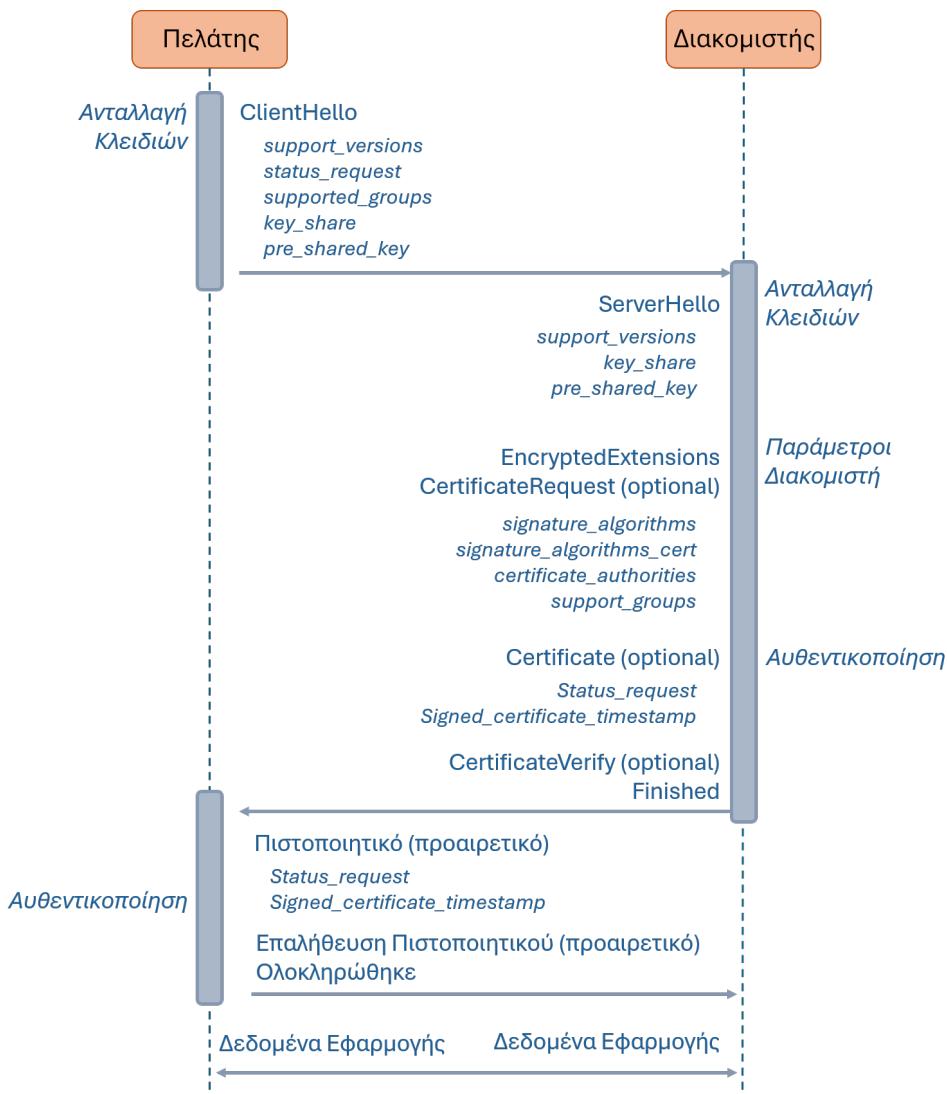
2. Παράμετροι διακομιστή – ο διακομιστής στέλνει δύο μηνύματα για να καθορίσει τις παραμέτρους διακομιστή (αναλύεται περαιτέρω στην Ενότητα 10.2.2.1.2):

- EncryptedExtensions (υποχρεωτικό): Περιέχει επεκτάσεις που μπορούν να προστατευτούν (κρυπτογραφηθούν), δηλαδή οποιεσδήποτε δεν χρειάζονται για τη δημιουργία του κρυπτογραφικού υλικού.
- CertificateRequest (προαιρετικό): Εάν απαιτείται η αυθεντικοποίηση πελάτη βάσει πιστοποιητικού, τότε ο διακομιστής στέλνει αυτό το μήνυμα το οποίο περιέχει τις επιθυμητές παραμέτρους για αυτό το πιστοποιητικό. Αυτό το μήνυμα παραλείπεται εάν δεν είναι απαραίτητη η αυθεντικοποίηση πελάτη.

3. Αυθεντικοποίηση (αναλύεται περαιτέρω στην Ενότητα 10.2.2.1.3):

- Ο διακομιστής στέλνει τα ακόλουθα μηνύματα αυθεντικοποίησης:
 - Certificate: Ο διακομιστής στέλνει το ψηφιακό του πιστοποιητικό το οποίο θα χρησιμοποιηθεί για την αυθεντικοποίησή του. Το πιστοποιητικό αποστέλλεται κάθε φορά που η συμφωνημένη μέθοδος ανταλλαγής κλειδιών χρησιμοποιεί πιστοποιητικά για αυθεντικοποίηση (παράδειγμα μηχανισμού που δεν απαιτεί την αποστολή πιστοποιητικού είναι η χρήση pre-shared keys όπου η αυθεντικοποίηση γίνεται με τη χρήση προ-διαμοιρασμένων κλειδιών).
 - CertificateVerify: Χρησιμοποιείται για να παρέχει ρητή απόδειξη ότι οι συμμετέχοντες διαθέτουν το ίδιωτικό κλειδί που αντιστοιχεί στο πιστοποιητικό τους. Το μήνυμα CertificateVerify παρέχει επίσης ακεραιότητα για τα μηνύματα της χειραψίας που ανταλλάχθηκαν μέχρι αυτό το σημείο, με τη χρήση υπογραφής η οποία δημιουργείται με το ίδιωτικό κλειδί που αντιστοιχεί στο δημόσιο κλειδί στο μήνυμα Certificate. Οι διακομιστές πρέπει να στείλουν το CertificateVerify όταν γίνεται αυθεντικοποίηση με τη χρήση πιστοποιητικού. Εάν ο διακομιστής δεν αυθεντικοποιείται βάσει πιστοποιητικού, το μήνυμα παραλείπεται.
 - Finished: Πρόκειται για το τελικό μήνυμα της αυθεντικοποίησης. Είναι απαραίτητο για την επιβεβαίωση της αυθεντικότητας της χειραψίας και των κρυπτογραφικών κλειδιών που έχουν υπολογιστεί. Περιλαμβάνει ένα HMAC (Keyed-Hash Message Authentication Code) για το σύνολο των μηνυμάτων της χειραψίας.
- Ο πελάτης απαντά με τα δικά του μηνύματα Certificate, CertificateVerify και Finished. Το μήνυμα Certificate παραλείπεται εάν ο διακομιστής δεν έστειλε μήνυμα CertificateRequest. Το μήνυμα CertificateVerify παραλείπεται εάν δε γίνεται αυθεντικοποίηση του πελάτη με πιστοποιητικό.

Με την επιτυχή αποστολή και αποδοχή των μηνυμάτων Finished από τις δύο πλευρές, ο πελάτης και ο διακομιστής μπορούν πλέον να ανταλλάξουν δεδομένα εφαρμογής με προστασία της εμπιστευτικότητας και της ακεραιότητάς τους.



Σχήμα 10.2: Πρωτόκολλο χειραψίας στο TLS 1.3.

10.2.2.1.1 Ανταλλαγή Κλειδιών

Τα μηνύματα ανταλλαγής κλειδιών, ClientHello και ServerHello, καθορίζουν τις δυνατότητες του πελάτη και του διακομιστή αναφορικά με την προστασία των μηνυμάτων και δημιουργούν κοινόχρηστα κρυπτογραφικά κλειδιά που χρησιμοποιούνται για την προστασία της ανταλλαγής των υπόλοιπων μηνυμάτων του πρωτοκόλλου χειραψίας και των δεδομένων της εφαρμογής.

Το πρωτόκολλο χειραψίας στο TLS ξεκινά με τον πελάτη να στέλνει ένα μήνυμα ClientHello στον διακομιστή. Αυτό το μήνυμα περιέχει τα ακόλουθα πεδία:

- **random**: 32 bytes που δημιουργούνται από μια ασφαλή γεννήτρια τυχαίων αριθμών.
 - **cipher_suites**: μια λίστα με τους αλγορίθμους που υποστηρίζονται από τον πελάτη, από τους οποίους θα επιλεγεί: (α) αυτός που θα χρησιμοποιηθεί για το Πρωτόκολλο Εγγραφής (συμπεριλαμβανομένου

του μήκους του μυστικού κλειδιού) και (β) η συνάρτηση σύνοψης που θα χρησιμοποιηθεί με τη συνάρτηση HKDF (HMAC-based Extract-and-Expand Key Derivation Function [5]) με φθίνουσα σειρά προτίμησης του πελάτη.

- **extensions:** Οι επεκτάσεις διευκολύνουν την προσθήκη νέων χαρακτηριστικών στο πρωτόκολλο TLS με ελάχιστο αντίκτυπο σε υπάρχουσες υλοποιήσεις. Στις επεκτάσεις που μπορεί να περιέχει το μήνυμα ClientHello περιλαμβάνονται οι ακόλουθες:
 - **supported_versions** (υποχρεωτικό): υποδεικνύει ποιες εκδόσεις του TLS υποστηρίζει ο πελάτης με φθίνουσα σειρά προτίμησης χρήσης αυτών.
 - **status_request**: υποδεικνύει ότι ο πελάτης θέλει να χρησιμοποιήσει ένα πρωτόκολλο ελέγχου κατάστασης πιστοποιητικού, όπως είναι το Ηλεκτρονικό Πρωτόκολλο Κατάστασης Πιστοποιητικού – ΗΠΚΠ (Ενότητα 7.1.3.2) [6]. Ο διακομιστής ενδέχεται να μην συμφωνήσει να το χρησιμοποιήσει.
 - **pre_shared_key**: Ένα προ-διαμοιρασμένο κλειδί (Pre-Shared Key – PSK) είναι ένα κοινόχρηστο μυστικό κλειδί το οποίο οι δύο πλευρές τυπικά μοιράστηκαν σε προηγούμενη επικοινωνία χρησιμοποιώντας κάποιο ασφαλές κανάλι. Το PSK μπορεί να δημιουργηθεί κατά την εκτέλεση του πρωτοκόλλου χειραψίας TLS και στη συνέχεια να χρησιμοποιηθεί για τη δημιουργία μιας νέας σύνδεσης σε μια επόμενη εκτέλεση του πρωτοκόλλου χειραψίας. Αυτό ονομάζεται επανάληψη συνεδρίας με ένα PSK (session resumption with a PSK). Μόλις ολοκληρωθεί η εκτέλεση του πρωτοκόλλου χειραψίας, ο διακομιστής μπορεί να στείλει στον πελάτη ένα αναγνωριστικό (ID) PSK που αντιστοιχεί σε ένα μοναδικό κλειδί που προέρχεται από την αρχική χειραψία.
 - **cookie**: Όταν ένας διακομιστής στέλνει ένα μήνυμα HelloRetryRequest³, μπορεί να συμπεριλάβει αυτήν την επέκταση ώστε να αναγκάσει τον πελάτη να αποδείξει πως είναι προσβάσιμος στην διεύθυνση του δικτύου που χρησιμοποιεί – ενέργεια η οποία παρέχει κάποια προστασία από επιθέσεις άρνησης υπηρεσίας (Denial of Service – DoS). Όταν ο πελάτης στέλνει ένα νέο μήνυμα ClientHello, πρέπει να αντιγράψει τα περιεχόμενα που έλαβε στο HelloRetryRequest σε μια επέκταση cookie στο νέο μήνυμα ClientHello.

Ο διακομιστής, εάν είναι σε θέση να διαπραγματευτεί ένα αποδεκτό σύνολο παραμέτρων χειραψίας, απαντά στο μήνυμα ClientHello του πελάτη με ένα μήνυμα ServerHello. Αυτό το μήνυμα περιέχει τα ακόλουθα πεδία:

- **cipher_suite**: Περιέχει τον αλγόριθμο κρυπτογράφησης που έχει επιλεγεί από τον διακομιστή από τη λίστα που δόθηκε από τον πελάτη με το ClientHello.cipher_suites.
- **extensions**: Περιέχει επεκτάσεις που απαιτούνται για τη δημιουργία του ασφαλούς καναλιού επικοινωνίας και τη διαπραγμάτευση της έκδοσης του πρωτοκόλλου. Οι επεκτάσεις που μπορεί να περιέχει το ServerHello περιλαμβάνουν τα εξής:
 - **supported_versions** (υποχρεωτικό): Υποδεικνύει ποια έκδοση του TLS χρησιμοποιεί ο διακομιστής. Το μήνυμα ServerHello πρέπει να περιέχει αυτήν την επέκταση.
 - **key_share**: Περιέχει τις κρυπτογραφικές παραμέτρους που θέλει να χρησιμοποιήσει ο διακομιστής σε αυτή τη διαδικασία ανταλλαγής κλειδιών.
 - **pre_shared_key**: Περιέχει το προ-διαμοιρασμένο κλειδί (pre-shared key) που συμφώνησε να χρησιμοποιήσει ο διακομιστής.

³Ο διακομιστής στέλνει ένα μήνυμα HelloRetryRequest ως απάντηση σε ένα μήνυμα ClientHello, για να ζητήσει από τον πελάτη να στείλει ξανά το μήνυμα ClientHello με τροποποιημένες παραμέτρους, γιατί το αρχικό ClientHello δεν πληροί τις απαιτήσεις του διακομιστή για τη σύναψη της ασφαλούς σύνδεσης.

10.2.2.1.2 Παράμετροι Διακομιστή

Αφού ο διακομιστής στέλει ένα μήνυμα ServerHello στον πελάτη, στέλνει δύο επιπλέον μηνύματα για να καθορίσει τις παραμέτρους διακομιστή: EncryptedExtensions και CertificateRequest.

- **EncryptedExtensions** (υποχρεωτικό): Αυτό είναι το πρώτο μήνυμα που στέλνει ο διακομιστής κρυπτογραφημένο με τα κλειδιά που δημιουργούνται ως αποτέλεσμα αυτής της χειραψίας. Το μήνυμα EncryptedExtensions περιέχει επεκτάσεις που πρέπει να προστατευτούν με κρυπτογράφηση. Αυτές οι επεκτάσεις δεν σχετίζονται με τη δημιουργία των κρυπτογραφικών παραμέτρων ή την παραγωγή κρυπτογραφικού υλικού.
- **CertificateRequest**: Αποστέλλεται όταν ο διακομιστής επιθυμεί αυθεντικοποίηση του πελάτη με χρήση ψηφιακού πιστοποιητικού. Περιλαμβάνει τα ακόλουθα πεδία:
 - **certificate_request_context**: Περιέχει ένα αναγνωριστικό που προσδιορίζει αυτό το αίτημα.
 - **extensions**: Περιέχει επεκτάσεις που περιγράφουν τις παραμέτρους του πιστοποιητικού που ζητήθηκε, ως εξής:
 - * **signature_algorithms**: Υποδεικνύει ποιοι αλγόριθμοι υπογραφής μπορούν να χρησιμοποιηθούν σε μηνύματα CertificateVerify. Το μήνυμα ServerHello πρέπει να περιέχει αυτήν την επέκταση.
 - * **signature_algorithms_cert**: Υποδεικνύει ποιοι αλγόριθμοι υπογραφής μπορούν να χρησιμοποιηθούν στις ψηφιακές υπογραφές.
 - * **certificateAuthorities**: Υποδεικνύει ποιες αρχές πιστοποιητικών αποδέχεται ο διακομιστής.

10.2.2.1.3 Αυθεντικοποίηση

Τα τρία τελευταία μηνύματα που ανταλλάσσονται ο διακομιστής και ο πελάτης σε μια χειραψία TLS είναι τα ακόλουθα:

- **Certificate**: Περιέχει το πιστοποιητικό αυθεντικοποίησης πελάτη/διακομιστή και τα υπόλοιπα πιστοποιητικά (αυτά των αρχών πιστοποίησης) στην αλυσίδα πιστοποίησης. Το μήνυμα Certificate περιλαμβάνει το πεδίο **certificate_list** το οποίο περιέχει μια ακολουθία δομών CertificateEntry, καθεμία από τις οποίες περιέχει ένα μόνο πιστοποιητικό. Επιπλέον περιλαμβάνει ένα σύνολο επεκτάσεων, όπως είναι η επέκταση **status_request**, με την οποία ο πελάτης (ή διακομιστής) μπορεί να ζητήσει από τον διακομιστή (έναν πελάτη αντίστοιχα) να προσκομίσει μαζί με το πιστοποιητικό του και μια απάντηση ΗΠΚΠ (Ηλεκτρονικό Πρωτόκολλο Κατάστασης Πιστοποιητικού, βλέπε Ενότητα 7.1.3.2), για την κατάσταση του πιστοποιητικού του.
- **CertificateVerify**: Αποτελεί μια υπογραφή σε όλα τα μηνύματα της χειραψίας χρησιμοποιώντας το ιδιωτικό κλειδί που αντιστοιχεί στο δημόσιο κλειδί στο μήνυμα Certificate. Το πεδίο αυτό αποδεικνύει ότι ο πελάτης ή ο διακομιστής αντίστοιχα, έχει το ιδιωτικό κλειδί που αντιστοιχεί στο πιστοποιητικό του. Περιλαμβάνει τα ακόλουθα πεδία:
 - **algorithm**: Περιέχει τον αλγόριθμο υπογραφής που χρησιμοποιείται.
 - **signature**: Περιέχει την ψηφιακή υπογραφή που δημιουργήθηκε από τον πελάτη ή τον διακομιστή χρησιμοποιώντας τον αλγόριθμο που ορίστηκε στο πεδίο **algorithm**.

- **Finished:** Περιέχει έναν κώδικα αυθεντικοποίησης μηνύματος (MAC) στο σύνολο των μηνυμάτων της χειραψίας. Μόλις ο πελάτης και ο διακομιστής επαληθεύσουν τα μηνύματα **Finished** που έχουν λάβει από την άλλη πλευρά, μπορούν να στείλουν και να λάβουν δεδομένα εφαρμογής μέσω του ασφαλούς καναλιού επικοινωνίας που έχουν δημιουργήσει.

10.2.2.2 Πρωτόκολλο Εγγραφής TLS

Το Πρωτόκολλο Εγγραφής TLS (TLS Record Protocol) [4], χρησιμοποιεί τις παραμέτρους που καθορίζονται από το πρωτόκολλο χειραψίας για την προστασία των δεδομένων που ανταλλάσσονται μεταξύ των δύο πλευρών. Παρέχει ένα ασφαλές κανάλι μεταξύ ενός πελάτη και ενός διακομιστή εξασφαλίζοντας τα ακόλουθα:

- **Εμπιστευτικότητα:** Τα δεδομένα κρυπτογραφούνται με τη χρήση κρυπτογραφίας συμμετρικού κλειδιού. Τα κλειδιά για αυτήν την κρυπτογράφηση δημιουργούνται αποκλειστικά για κάθε κατεύθυνση σύνδεσης και βασίζονται σε ένα μυστικό το οποίο διαπραγματεύονται οι δύο πλευρές με τη χρήση του πρωτοκόλλου χειραψίας.
- **Ακεραιότητα και αυθεντικοποίηση μηνύματος:** Η ανταλλαγή των μηνυμάτων μεταξύ των δύο πλευρών στην επικοινωνία περιλαμβάνει έναν έλεγχο ακεραιότητας και αυθεντικοποίησης κάθε μηνύματος με τη χρήση Κώδικα Αυθεντικοποίησης Μηνύματος (MAC).

Το πρωτόκολλο εγγραφής διαιρεί τα δεδομένα ενός μηνύματος σε μπλοκ, τα οποία εδώ ονομάζονται εγγραφές (record), καθεμία από τις οποίες προστατεύεται ανεξάρτητα χρησιμοποιώντας τα απαραίτητα κλειδιά κρυπτογράφησης. Πιο συγκεκριμένα, το πρωτόκολλο εγγραφής κατακερματίζει τα δεδομένα ενός μηνύματος (αρχικό μήνυμα) σε **TLSPlaintext** εγγραφές, καθεμία από τις οποίες περιέχει τμήματα δεδομένων που περιλαμβάνουν έως 2^{14} bytes. Στη συνέχεια, τα μηνύματα προστατεύονται μέσω μιας διαδικασίας που περιλαμβάνει κρυπτογράφηση, αυθεντικοποίηση και προαιρετική πλήρωση, και ενθυλακώνονται σε εγγραφές **TLSCiphertext** προκειμένου να μεταδοθούν προς την άλλη πλευρά. Το αρχικό μήνυμα μπορεί να είναι διαφόρων τύπων, όπως μήνυμα χειραψίας, δεδομένα εφαρμογής ή κάποια ειδοποίηση (alert), το καθένα με τη δική του δομή και περιεχόμενο.

Η διαδικασία προετοιμασίας και αποστολής μηνυμάτων περιλαμβάνει τα ακόλουθα βήματα:

- Το αρχικό μήνυμα **TLSPlaintext** κρυπτογραφείται χρησιμοποιώντας έναν αλγόριθμο AEAD (Authenticated Encryption with Associated Data). Το TLS 1.3 επιβάλλει τη χρήση κρυπτογράφησης AEAD για προστασία μηνυμάτων. Για τη διαδικασία αυτή χρησιμοποιείται ένα κλειδί κρυπτογράφησης για κάθε κατεύθυνση (**client_write_key** ή **server_write_key**).
- Το κρυπτογραφημένο κείμενο τοποθετείται σε μια δομή **TLSCiphertext**, η οποία περιλαμβάνει τα ακόλουθα πεδία:
 - **opaque_type:** Αυτό το πεδίο ορίζεται πάντα σε **application_data** για συμβατότητα με προηγούμενες εκδόσεις του TLS. Ο πραγματικός τύπος περιεχομένου του μηνύματος αποθηκεύεται στο πεδίο **TLSInnerPlaintext.type** μετά την αποκρυπτογράφηση.
 - **legacy_record_version:** Αυτό το πεδίο ορίζεται πάντα σε **0x0303**. Τα **TLSCiphertext** μηνύματα στο TLS1.3 δεν δημιουργούνται παρά μόνο μετά τη διαπραγμάτευση του TLS 1.3, επομένως δεν υπάρχουν προβλήματα σχετικά με τη συμβατότητα με προηγούμενες εκδόσεις.
 - **length:** Το μήκος του **TLSCiphertext.encrypted_record**. Περιλαμβάνει το μήκος των δεδομένων που κρυπτογραφούνται και περιλαμβάνονται στο πεδίο **encrypted_record**, όπως τα δεδομένα της εφαρμογής, το μήκος του MAC και το μήκος της πλήρωσης (padding).
 - **encrypted_record:** Περιέχει το αποτέλεσμα της κρυπτογράφησης του αρχικού μηνύματος **TLSInnerPlaintext** με τον αλγόριθμο AEAD.

Η πλήρωση, αποτελούμενη από bytes με τιμή 0x00, προστίθεται στην εγγραφή **TLSCiphertext** πριν από την κρυπτογράφηση, επιτρέποντας έτσι στους αποστολείς των μηνυμάτων να κρύβουν το πραγματικό μέγεθος του μηνύματος, και να ενισχύουν με αυτόν τον τρόπο το απόρρητο και την ασφάλεια. Το μήκος της πλήρωσης πρέπει να λαμβάνεται υπόψη κατά τον υπολογισμό του συνολικού μήκους της εγγραφής **TLSCiphertext**.

10.3 IPSec

Το IPSec (Internet Protocol Security) [7, 8, 9, 10] αποτελεί πρακτικά ένα πλαίσιο που μας βοηθά να προστατεύουμε την κίνηση στο Διαδίκτυο. Η προστασία των δεδομένων λαμβάνει χώρα στο επίπεδο δικτύου της στοίβας TCP/IP, προσφέροντας έτσι προστασία στο επίπεδο δικτύου καθώς και σε δεδομένα ανώτερων επιπέδων. Με αυτόν τον τρόπο καλύπτει τα κενά ασφαλείας του πρωτοκόλλου IP (Internet Protocol). Συγκεκριμένα, παρέχει προστασία στην ανταλλαγή των IP δεδομενογράμματα μεταξύ των συμμετεχουσών συσκευών IPSec εξασφαλίζοντας τα ακόλουθα:

- **Εμπιστευτικότητα:** Με την κρυπτογράφηση των δεδομένων, κανείς εκτός από τον αποστολέα και τον παραλήπτη δεν μπορεί να διαβάσει ή να επεξεργαστεί αυτά που ανταλλάσσονται οι δύο πλευρές.
- **Ακεραιότητα:** Με την παραγωγή συνόψεων, ο αποστολέας και ο παραλήπτης μπορούν να ελέγξουν εάν έχουν γίνει αλλαγές στα IP δεδομενογράμματα που ανταλλάσσονται.
- **Αυθεντικοποίηση:** Ο αποστολέας και ο παραλήπτης επιβεβαιώνουν την ταυτότητα της άλλης πλευράς.
- **Anti-replay:** Ακόμα κι αν ένα δεδομενόγραμμα είναι κρυπτογραφημένο και αυθεντικοποιημένο, ένας εισβολέας θα μπορούσε να προσπαθήσει να το υποκλέψει και να το στείλει ξανά (replay attack). Χρησιμοποιώντας ακολουθιακούς αριθμούς, το IPSec αποτρέπει τη επαναμετάδοση δεδομενογραμμάτων.

Το IPSec αποτελεί ένα αρκετά πολύπλοκο πλαίσιο, ωστόσο αυτό είναι αποτέλεσμα των πολλών διαφορετικών επιλογών χρήστης και της εφαρμογής του σε διαφορετικούς δικτυακούς κόμβους, όπως οι δρομολογητές, τείχη προστασίας, κεντρικούς υπολογιστές και διακομιστές. Παραδείγματα χρήσης περιλαμβάνουν τα ακόλουθα:

- Μεταξύ δύο δρομολογητών για τη δημιουργία ενός Εικονικού Ιδιωτικού Δικτύου (VPN – Virtual Private Network), δηλαδή την επίτευξη ασφαλούς επικοινωνίας μεταξύ δύο απομακρυσμένων τοποθεσιών (υποδικτύων) μέσω του ανασφαλούς διαδικτύου, παρέχοντας μια ιδιωτική σήραγγα για τα δεδομένα και τις επικοινωνίες πάνω από δημόσια δίκτυα.
- Ανάμεσα σε ένα τείχος προστασίας και έναν υπολογιστή-πελάτη για την ασφαλή απομακρυσμένη πρόσβαση σε ελεγχόμενο, προστατευμένο δίκτυο.
- Μεταξύ δύο διακομιστών για την προστασία των δεδομένων που ανταλλάσσονται με τη χρήση ενός πρωτοκόλλου το οποίο δε μεριμνά για την προστασία των δεδομένων.

Για την προστασία των δεδομενογραμμάτων IP με τη χρήση του IPSec, απαιτείται απαιτείται η δημιουργία μιας σύνδεσης IPSec με τη χρήση ενός πρωτοκόλλου που ονομάζεται IKE (Internet Key Exchange) [11, 12].

Υπάρχουν δύο φάσεις για την κατασκευή μιας σήραγγας IPSec:

- 1η φάση IKE
- 2η φάση IKE

Στη 1η φάση IKE, οι δύο κόμβοι διαπραγματεύονται σχετικά με την κρυπτογράφηση, την αυθεντικοποίηση, την παραγωγή συνόψεων και άλλα πρωτόκολλα που θέλουν να χρησιμοποιήσουν, καθώς και ορισμένες άλλες παραμέτρους που απαιτούνται. Σε αυτή τη φάση, δημιουργείται μια συνεδρία ISAKMP (Internet Security Association and Key Management Protocol). Αυτό ονομάζεται επίσης σήραγγα ISAKMP ή σήραγγα 1ης φάσης IKE. Το σύνολο των παραμέτρων που θα χρησιμοποιήσουν οι δύο πλευρές ονομάζεται Συσχετισμός Ασφαλείας (Security Association – SA). Οι Συσχετισμοί Ασφαλείας αναλύονται στην Ενότητα 10.3.3. Στο Σχήμα 10.3 παρουσιάζεται ένα παράδειγμα δύο δρομολογητών που έχουν δημιουργήσει τη σήραγγα 1ης φάσης του IKE.



Σχήμα 10.3: Σήραγγα 1ης φάσης IKE.

Η σήραγγα 1ης φάσης του IKE χρησιμοποιείται μόνο για διαχείριση κίνησης, ως μία ασφαλής μέθοδος για τη δημιουργία της δεύτερης σήραγγας που ονομάζεται σήραγγα 2ης φάσης του IKE ή σήραγγα IPSec και για διαχείριση της κυκλοφορίας, όπως την αποστολή των keepalive μηνυμάτων. Στο Σχήμα 10.4 απεικονίζεται η κατάσταση δύο δρομολογητών που ολοκλήρωσαν τη 2η φάση του IKE.



Σχήμα 10.4: Σήραγγα 2ης φάσης IKE.

Με την ολοκλήρωση της 2ης φάσης του IKE, έχουμε μια σήραγγα IPSec που μπορούμε να χρησιμοποιήσουμε για την προστασία των δεδομένων των δύο πλευρών. Αυτά τα δεδομένα θα σταλούν μέσω της σήραγγας της 2ης φάσης IKE όπως απεικονίζεται στο Σχήμα 10.5.



Σχήμα 10.5: Ανταλλαγή δεδομένων με τη χρήση σήραγγας 2ης φάσης IKE.

Το IKE δημιουργεί τις σήραγγες για τις δύο πλευρές, αλλά δεν επαληθεύει την πηγή ή την ακεραιότητα των δεδομένων, ούτε κρυπτογραφεί τα δεδομένα χρήστη. Οι υπηρεσίες αυτές παρέχονται από δύο άλλα πρωτόκολλα:

- Κεφαλίδα Αυθεντικοποίησης (Authentication Header – AH) [8]: Το AH Προστατεύει το μεγαλύτερο μέρος του δεδομενογράμματος IP και παρέχει αυθεντικοποίηση των δεδομενογραμμάτων βάσει

χρήστη ή IP διεύθυνσης πηγής, ακεραιότητα των δεδομένων και (προαιρετικά) προστασία από επαναλήψεις. Δεν κρυπτογραφεί το δεδομενόγραμμα ή μέρος αυτού.

Το AH ορίζει μια κεφαλίδα που προστίθεται στο IP δεδομενόγραμμα που περιλαμβάνει τα εξής πεδία:

- Μήκος δεδομένων.
- Ένα Δείκτη Παραμέτρων Ασφαλείας (Security Parameter Index – SPI) ο οποίος λειτουργεί ως αναγνωριστικό των αλγορίθμων και των κλειδιών που θα χρησιμοποιηθούν από το IPSec.
- Έναν ακολουθιακό αριθμό (sequence number).
- Δεδομένα αυθεντικοποίησης: ένας κώδικας αυθεντικοποίησης μηνύματος (MAC) ο οποίος υπολογίζεται σε όλα τα πεδία της κεφαλίδας εκτός από το Time-to-Live (TTL) και σε όλα τα δεδομένα ή το εσωτερικό IP δεδομενόγραμμα.
- Ενθυλάκωση Ωφέλιμου Φορτίου Ασφαλείας (Encapsulated Security Payload – ESP): Το ESP παρέχει εμπιστευτικότητα με την κρυπτογράφηση του μεγαλύτερου τμήματος του δεδομενογράμματος καθώς και αυθεντικοποίηση και ακεραιότητα των δεδομένων. Συγκεκριμένα, το ESP παρέχει εμπιστευτικότητα μόνο στα δεδομένα ωφέλιμου φορτίου IP. Παρέχει επίσης τις υπηρεσίες του πρωτοκόλλου AH. Ωστόσο, σε αντίθεση με το AH, το ESP δεν προστατεύει την κεφαλίδα IP του δεδομενογράμματος. Παρόλα αυτά, λόγω της υποστήριξης της κρυπτογράφησης των δεδομένων, το ESP είναι η πιο δημοφιλής επιλογή στις μέρες μας.

Ορίζει μια κεφαλίδα που περιλαμβάνει:

- Έναν Δείκτη Παραμέτρων Ασφαλείας (Security Parameter Index – SPI).
- Έναν ακολουθιακό αριθμό.

και κάποια πεδία που ακολουθούν τα δεδομένα του ωφέλιμου φορτίου (trailing fields):

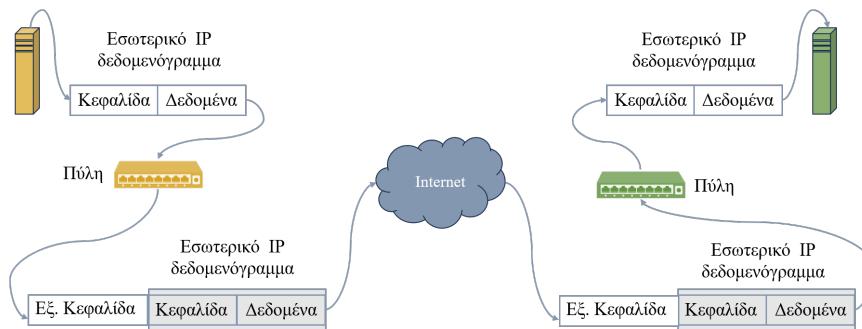
- Πλήρωση (padding), εάν χρειάζεται από τον αλγόριθμο κρυπτογράφησης.
- Το μήκος της πλήρωσης.
- Δεδομένα αυθεντικοποίησης (εάν υπάρχουν), δηλ. το MAC.

Και τα δύο πρωτόκολλα υποστηρίζουν δύο διαφορετικούς τρόπους λειτουργίας, όπως αναλύονται στις ακόλουθες ενότητες.

10.3.1 Λειτουργία Σήραγγας

Στη λειτουργία σήραγγας (Tunnel Mode) ολόκληρο το αρχικό IP δεδομενόγραμμα ενθυλακώνεται σε ένα νέο IP δεδομενόγραμμα και γίνεται το ωφέλιμο φορτίο αυτού. Στο νέο IP δεδομενόγραμμα προστίθεται μια νέα κεφαλίδα IP η οποία μεταφέρει πληροφορίες που αφορούν στους δικτυακούς κόμβους που υλοποιούν αυτή τη λειτουργία IPSec, ενώ το νέο ωφέλιμο φορτίο προετοιμάζεται κατάλληλα, βάσει των απαιτήσεων του πρωτοκόλλου (AH ή ESP) που έχει επιλεγεί, ώστε να προστατευτεί κατά τη διακίνηση του μέσα από δημόσια ή μη ασφαλή δίκτυα, όπως απεικονίζεται και στο Σχήμα 10.6. Στο συγκεκριμένο παράδειγμα, η λειτουργία της σήραγγας υλοποιείται μεταξύ των δύο πυλών ασφαλείας που ανήκουν σε διαφορετικά δίκτυα και προστατεύει τα δεδομενογράμματα που διακινούνται μεταξύ αυτών.

Ένα πλεονέκτημα αυτής της λειτουργίας είναι ότι καθιστά πολύ εύκολη τη δημιουργία μιας ασφαλούς σύνδεσης, με τη μορφή της σήραγγας, μεταξύ δύο πυλών ασφαλείας που χρησιμοποιούν το IPSec. Αυτές οι πύλες IPSec με τη σειρά τους μπορούν να συνδέσουν δύο διαφορετικά δίκτυα με ασφάλεια. Η χρήση ασφαλών διακομιστών IPSec όπως απεικονίζεται στο Σχήμα 10.6 μπορεί να είναι πολύ χρήσιμη για τη σύνδεση δύο απομακρυσμένων δικτύων χρησιμοποιώντας μια κρυπτογραφημένη σύνδεση, δημιουργώντας έτσι ένα εικονικό ιδιωτικό δίκτυο (VPN).



Σχήμα 10.6: Χρήση του IPSec μεταξύ δύο πυλών, με τη χρήση της λειτουργίας σήραγγας.

Η διαδικασία που χρησιμοποιείται από το IPSec για την ενθυλάκωση της αρχικής κεφαλίδας IP διαφέρει ανάλογα με το εάν χρησιμοποιείται η λειτουργία σήραγγας AH ή η λειτουργία σήραγγας ESP:

- Το αρχικό δεδομενόγραμμα ενθυλακώνεται σε ένα νέο δεδομενόγραμμα IP (τόσο η κεφαλίδα IP όσο και το ωφέλιμο φορτίο).
- Στην περίπτωση της λειτουργίας σήραγγας AH, προστίθεται μια κεφαλίδα AH και μια νέα κεφαλίδα IP. Για τη λειτουργία σήραγγας ESP, προστίθενται μια κεφαλίδα ESP, μια νέα κεφαλίδα IP, ένα τρέιλερ ESP και ένα τρέιλερ αυθεντικοποίησης ESP, όπως φαίνεται στο Σχήμα 10.7.

Σε κάθε περίπτωση, η εξωτερική κεφαλίδα IP καθορίζει ποιός θα επεξεργαστεί αυτό το δεδομενόγραμμα που προστατεύεται από το IPSec, ενώ η εσωτερική κεφαλίδα IP καθορίζει ποιός είναι ο τελικός παραλήπτης αυτού του δεδομενογράμματος.

AH με χρήση λειτουργίας σήραγγας:

Εξωτερική IP κεφαλίδα	AH κεφαλίδα Len, SPI, seqno, MAC	Εσωτερική IP κεφαλίδα	Ωφέλιμο Φορτίο (π.χ. TCP, UDP, ICMP)
↔ Κάλυψη MAC - όλα τα αμετάβλητα πεδία			

ESP με χρήση λειτουργίας σήραγγας:

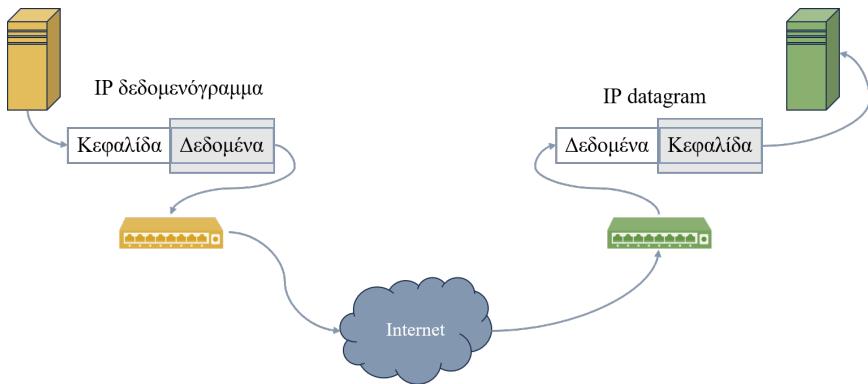
Εξωτερική IP κεφαλίδα	ESP κεφαλίδα SPI, seqno	Εσωτερική IP κεφαλίδα	Ωφέλιμο Φορτίο (π.χ. TCP, UDP, ICMP)	ESP trlr	ESP auth
↔ Κάλυψη MAC					
↔ Κάλυψη κρυπτογράφησης					

Σχήμα 10.7: AH και ESP με χρήση λειτουργίας σήραγγας.

Όταν χρησιμοποιείται η λειτουργία σήραγγας με AH, ο μηχανισμός αυθεντικοποίησης καλύπτει όλα τα αμετάβλητα πεδία του δεδομενογράμματος. Άλλα όταν χρησιμοποιείται η λειτουργία σήραγγας με ESP, ο μηχανισμός αυθεντικοποίησης καλύπτει το ενθυλακωμένο δεδομενόγραμμα μεταξύ της κεφαλίδας ESP και του τρέιλερ ESP, καλύπτοντας έτσι όλο το αρχικό IP δεδομενόγραμμα. Ωστόσο, το μεγάλο πλεονέκτημα της χρήσης της σήραγγας με ESP είναι πως σε αυτή τη λειτουργία κρυπτογραφείται όλο το αρχικό IP δεδομενόγραμμα, συμπεριλαμβανομένης της αρχικής κεφαλίδας, δίνοντας έτσι τη δυνατότητα στον αποστολέα να αποκρύψει λεπτομέρειες της κεφαλίδας IP, μεταξύ των οποίων και η IP διεύθυνσή του.

10.3.2 Λειτουργία Μεταφοράς

Στη λειτουργία μεταφοράς (Transport Mode) η αρχική κεφαλίδα διατηρείται στο τροποποιημένο δεδομενόγραμμα, χωρίς την προσθήκη νέας κεφαλίδας όπως γίνεται στο ESP. Με άλλα λόγια, τα δεδομένα ωφέλιμου φορτίου που μεταδίδονται μέσα στο αρχικό δεδομενόγραμμα IP προστατεύονται, αλλά όχι η κεφαλίδα IP. Στη λειτουργία μεταφοράς, η κρυπτογραφημένη κίνηση αποστέλλεται απευθείας μεταξύ δύο κεντρικών υπολογιστών που δημιουργησαν προηγουμένως μια ασφαλή σήραγγα IPSec, όπως απεικονίζεται στο Σχήμα 10.8.



Σχήμα 10.8: Χρήση του IPSec σε λειτουργία μεταφοράς.

Αφού δε χρησιμοποιείται νέα κεφαλίδα, η διαδικασία που χρησιμοποιείται από τη λειτουργία μεταφοράς είναι λιγότερο περίπλοκη από τη λειτουργία σήραγγας, όπως απεικονίζεται στο Σχήμα 10.9:

- Ανάλογα με το πρωτόκολλο που χρησιμοποιείται, δημιουργείται μια νέα κεφαλίδα AH ή ESP και εισάγεται ακριβώς μετά την αρχική κεφαλίδα IP.
- Για το πρωτόκολλο ESP, τόσο ένα τρέιλερ ESP όσο και ένα τρέιλερ αυθεντικοποίησης δημιουργούνται και προστίθενται μετά το αρχικό δεδομενόγραμμα.
- Όταν χρησιμοποιείται η λειτουργία μεταφοράς AH, ο μηχανισμός αυθεντικοποίησης εφαρμόζεται σε όλα τα αμετάβλητα πεδία του δεδομενογράμματος. Για τη λειτουργία μεταφοράς ESP, ο μηχανισμός αυθεντικοποίησης εφαρμόζεται σε όλο το αρχικό ωφέλιμο φορτίο δεδομενογράμματος (δηλ. χωρίς να συμπεριλαμβάνεται η κεφαλίδα IP) και κρυπτογραφείται.

Επομένως, όταν χρησιμοποιείται μέθοδος μεταφοράς με το ESP, το IPSec παρέχει υπηρεσίες ασφαλείας μόνο σε πρωτόκολλα ανώτερων επιπέδων, όχι στην κεφαλίδα του IP δεδομενογράμματος ή άλλες κεφαλίδες προέκτασης (extension header) [13] που βρίσκονται πριν την ESP κεφαλίδα. Αν χρησιμοποιείται AH η προστασία περιλαμβάνει συγκεκριμένα τμήματα της κεφαλίδας IP και των κεφαλίδων προέκτασης.

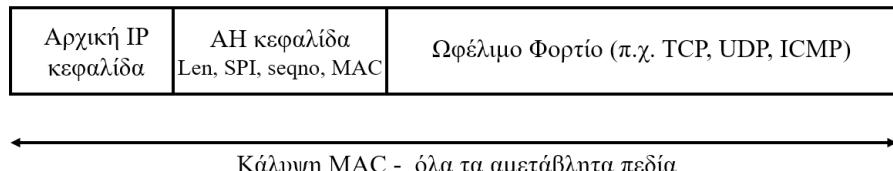
10.3.3 Συσχετισμοί Ασφαλείας

Το IPSec επιτρέπει στους χρήστες να καθορίσουν πότε και για ποιά δεδομένα θα χρησιμοποιηθεί καθώς και το είδος των υπηρεσιών που θα προσφέρονται από αυτό. Για παράδειγμα, κάποιος μπορεί να δημιουργήσει μια κρυπτογραφημένη σήραγγα για να μεταφέρει όλα τα δεδομένα που ανταλλάσσονται δύο πύλες, ή μια ξεχωριστή κρυπτογραφημένη σήραγγα για κάθε σύνδεση TCP μεταξύ δύο σταθμών που επικοινωνούν μέσω αυτών των πυλών.

Η δημιουργία μιας σύνδεσης με το IPSec γίνεται σύμφωνα με τα ακόλουθα γενικευμένα βήματα:

- Η μια πλευρά (ή συσκευή) αρχικοποιεί τη διαδικασία στέλνοντας μια αίτηση προς την άλλη για σύνδεση IPSec.

AH με χρήση λειτουργίας μεταφοράς (IPv4):



ESP με χρήση λειτουργίας μεταφοράς (IPv4):



Σχήμα 10.9: AH και ESP με χρήση λειτουργίας μεταφοράς.

- Οι δύο πλευρές συμφωνούν στους αλγορίθμους κρυπτογράφησης, στις συναρτήσεις σύνοψης και στα κρυπτογραφικά κλειδιά που θα χρησιμοποιήσουν.
- Η κάθε πλευρά δημιουργεί έναν Συσχετισμό Ασφαλείας (Security Association – SA) για κάθε κατεύθυνση της σύνδεσης.

Ως αποτέλεσμα της επιτυχούς δημιουργίας αυτής της σύνδεσης η κάθε πλευρά αυθεντικοποιεί, κατακερματίζει, κρυπτογραφεί και αποκρυπτογραφεί δεδομενογράμματα που ανταλλάσσονται, βάσει των παραμέτρων που έχουν οριστεί στον Συσχετισμό Ασφαλείας.

Ο Συσχετισμός Ασφαλείας (Security Association – SA) είναι μια θεμελιώδης έννοια που χρησιμοποιείται για τη δημιουργία και τη διαχείριση της ασφαλούς επικοινωνίας μεταξύ των δικτυακών συσκευών που υλοποιούν το IPSec, όπως σταθμοί, δρομολογητές ή πύλες ασφαλείας. Είναι ένα σύνολο παραμέτρων ασφαλείας και κλειδιών που διέπουν την κρυπτογράφηση, την αυθεντικοποίηση και την προστασία της ακεραιότητας των δεδομένων που ανταλλάσσονται μεταξύ αυτών των συσκευών.

Ένας Συσχετισμός Ασφαλείας αποτελείται από τα ακόλουθα βασικά στοιχεία:

- Δείκτης Παραμέτρων Ασφαλείας (Security Parameter Index – SPI): Το SPI είναι ένα μοναδικό αναγνωριστικό που βοηθά στη διάκριση ενός συσχετισμού ασφαλείας από έναν άλλο, ειδικά όταν υπάρχουν πολλαπλοί SAs μεταξύ του ίδιου ζεύγους συσκευών. Κάθε SA έχει το δικό του SPI και περιλαμβάνεται στις κεφαλίδες IPSec για να προσδιορίσει το κατάλληλο SA για την επεξεργασία των εισερχόμενων πακέτων.
- Πρωτόκολλο ασφαλείας (AH ή ESP): Το Security Association καθορίζει ποιο πρωτόκολλο IPSec θα χρησιμοποιηθεί για την ασφάλεια της κυκλοφορίας.
- Αλγόριθμοι κρυπτογράφησης και αυθεντικοποίησης: Το SA περιλαμβάνει τους αλγόριθμους και τα κλειδιά κρυπτογράφησης και αυθεντικοποίησης που θα χρησιμοποιηθούν για την προστασία των δεδομένων. Αυτοί οι αλγόριθμοι είναι το αποτέλεσμα της διαπραγμάτευσης που λαμβάνει χώρα κατά την εκτέλεση του IKE πρωτοκόλλου, το οποίο οδηγεί στη δημιουργία των SAs.
- Πληροφορίες διαχείρισης κλειδιών: Περιέχει λεπτομέρειες σχετικά με τον τρόπο δημιουργίας, διανομής και διαχείρισης των κλειδιών για κρυπτογράφηση και αυθεντικοποίηση.

- Διάρκεια ζωής: Ένας SA μπορεί να έχει καθορισμένη διάρκεια ζωής, με το πέρας της οποίας πρέπει οι δύο πλευρές να επαναδημιουργήσουν για να διασφαλιστεί ότι οι παράμετροι ασφαλείας παραμένουν ισχυρές.
- Επιλογέας κυκλοφορίας (Traffic Selector): Ορίζει το εύρος του SA, προσδιορίζοντας ποιες διευθύνσεις IP, θύρες και πρωτόκολλα προστατεύονται από την SA, καθώς και ποια κίνηση υπόκειται σε επεξεργασία IPSec.

Ένας συσχετισμός ασφαλείας μπορεί να χρησιμοποιεί το AH ή το ESP πρωτόκολλο αλλά όχι και τα δύο. Εάν χρησιμοποιούνται και τα δύο πρωτόκολλα (AH και ESP) τότε πρέπει να έχουμε δύο διαφορετικά SAs. Δύο SAs πρέπει να υπάρχουν και για μια αμφίδρομη επικοινωνία μεταξύ δύο κόμβων, μία για κάθε κατεύθυνση. Ένας συσχετισμός ασφαλείας αναγνωρίζεται μοναδικά από τρεις παραμέτρους:

- Τον δείκτη παραμέτρων ασφαλείας (SPI).
- Την IP διεύθυνση προορισμού.
- Το αναγνωριστικό για το πρωτόκολλο που χρησιμοποιείται: AH ή ESP.

Ο συσχετισμός ασφαλείας υποδεικνύει στον παραλήπτη των δεδομένων πως να αποκρυπτογραφήσει, πως να αυθεντικοποιήσει την πηγή του δεδομενογράμματος, ποια κλειδιά να χρησιμοποιήσει, ποιους αλγορίθμους, και πως να απαντήσει στον αποστολέα. Εάν χρησιμοποιούνται το AH και το ESP πρωτόκολλα μαζί τότε μια συσκευή που θα πάρει ένα δεδομενόγραμμα θα κάνει τα εξής:

- Θα αναγνωρίσει τα σχετικά SPI, SA, μυστικά κλειδιά και συνάρτηση σύνοψης.
- Θα υπολογίσει τη σύνοψη στο δεδομενόγραμμα για να αυθεντικοποιήσει την πηγή και να επαληθεύσει την ακεραιότητα των δεδομένων.
- Θα βρει τον σωστό αλγόριθμο κρυπτογράφησης και τα σχετικά κλειδιά.
- Θα αποκρυπτογραφήσει το μήνυμα.

Όπως έχει αναλυθεί παραπάνω, το IPSec, ανάλογα με το πρωτόκολλο που χρησιμοποιείται (AH ή ESP) και τον τρόπο χρήσης του (μεταφορά ή σήραγγα) μπορεί να παρέχει διαφορετικού τύπου προστασία στα IP δεδομενογράμματα. Όλα αυτά αποτυπώνονται στους αντίστοιχους SAs. Επομένως, το σύνολο των υπηρεσιών ασφαλείας που προσφέρονται από ένα SA εξαρτάται από:

- Το επιλεγμένο πρωτόκολλο ασφαλείας (AH ή ESP).
- Τον τρόπο χρήσης (μεταφορά ή σήραγγα).
- Τα άκρα του SA (host ή πύλη).
- Την επιλογή των προαιρετικών υπηρεσιών που προσφέρει το πρωτόκολλο.

Εάν οι ανάγκες για προστασία των δεδομένων δε καλύπτονται από έναν μόνο SA (ο οποίος όπως αναφέρθηκε μπορεί να υλοποιήσει μόνο ένα πρωτόκολλο), τότε απαιτείται η χρήση πολλαπλών SAs ή δέσμη SAs. Οι SAs που χρησιμοποιούνται μπορεί να τερματίζουν σε διαφορετικά τελικά σημεία (άκρα): ένας συσχετισμός ασφαλείας μπορεί να χρησιμοποιείται μεταξύ ενός σταθμού (host) και μιας πύλης, ενώ ένας δεύτερος να τερματίζει σε έναν άλλο σταθμό ή διακομιστή πίσω από αυτήν την πύλη, όπως αποτυπώνεται στα παραδείγματα που δίνονται στην επόμενη ενότητα.

10.3.4 Συνδυασμοί Συσχετισμών Ασφαλείας

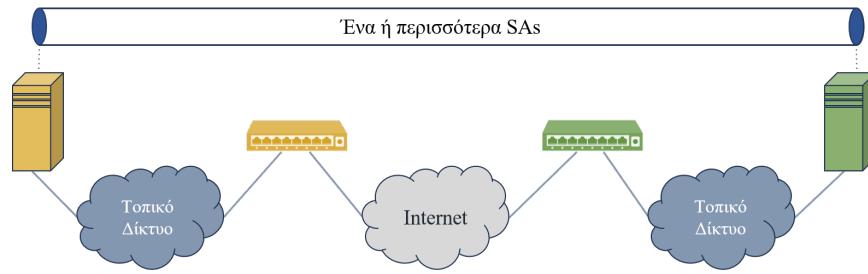
Σε αυτή την ενότητα παρουσιάζονται συνοπτικά παραδείγματα χρήσης του IPSec.

10.3.4.1 Παράδειγμα 1: Εφαρμογή του IPSec από Άκρο-σε-Άκρο

Εάν το IPSec χρησιμοποιείται μεταξύ δύο ακραίων σταθμών για την προστασία των μεταξύ τους επικοινωνιών, τότε αυτοί μπορούν να χρησιμοποιήσουν έναν ή περισσότερους συσχετισμούς ασφαλείας, με έναν από τους ακόλουθους συνδυασμούς:

- AH σε λειτουργία μεταφοράς.
- ESP σε λειτουργία μεταφοράς.
- Συνδυαστικά AH και ESP (π.χ. AH ακολουθούμενο από το ESP), και τα δύο σε λειτουργία μεταφοράς.
- Οποιοδήποτε από τα παραπάνω, σε μορφή σήραγγας με τη χρήση του AH ή ESP.

Η από άκρο-σε-άκρο προσέγγιση (Σχήμα 10.10) έχει το πλεονέκτημα πως εξασφαλίζει προστασία των μηνυμάτων από άκρο-σε-άκρο. Το μειονέκτημα ωστόσο αυτής της λύσης είναι πως το IPSec θα πρέπει να υλοποιηθεί σε όλους τους σταθμούς ξεχωριστά, και επομένως να δημιουργηθούν συσχετισμοί ασφάλειας για όλες τις επιμέρους συνδέσεις.



Σχήμα 10.10: IPsec από άκρο-σε-άκρο (υλοποίηση σε σταθμούς).

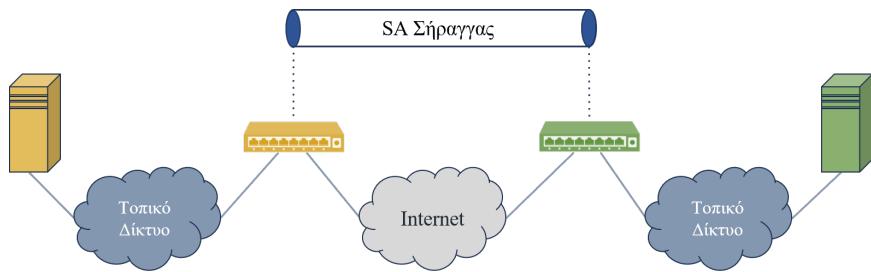
10.3.4.2 Παράδειγμα 2: Εφαρμογή του IPsec από Πύλη-σε-Πύλη

Σε αυτό το παράδειγμα το IPsec δεν υλοποιείται στους σταθμούς αλλά χρησιμοποιείται για να προστατεύσει τις επικοινωνίες μεταξύ δύο πυλών. Ένα τέτοιο ασφαλές κανάλι επιτρέπει την ανταλλαγή δεδομένων μεταξύ των δύο δικτύων προστατεύοντας τα δεδομένα όλων των σταθμών που βρίσκονται πίσω από αυτές τις πύλες (Σχήμα 10.11). Έτσι, απαιτείται μόνο ένας SA (για κάθε κατεύθυνση) για την χρήση του IPsec με τη μέθοδο της σήραγγας, η οποία χρησιμοποιεί το ESP.

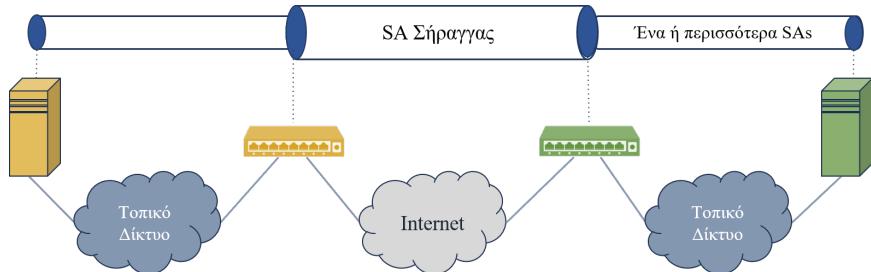
Το πλεονέκτημα αυτής της επιλογής είναι πως για την προστασία των επικοινωνιών μεταξύ των κόμβων δύο δικτύων, όπως είναι τα δίκτυα δύο παραρτημάτων ενός οργανισμού, απαιτείται η δημιουργία μιας μόνο σύνδεσης IPsec, αυτής μεταξύ των δύο πυλών. Το μειονέκτημα ωστόσο είναι πως δεν παρέχει ασφάλεια από άκρο-σε-άκρο, καθώς όσο το δεδομενόγραμμα διακινείται εντός των δικτύων, δεν προστατεύεται.

10.3.4.3 Παράδειγμα 3: Συνδυασμός των Παραδειγμάτων 1 και 2

Σε αυτό το παράδειγμα (βλέπε Σχήμα 10.12) υπάρχει σήραγγα από πύλη-σε-πύλη όπως στο Παράδειγμα 2, η οποία μεταφέρει κίνηση σταθμού-προς-σταθμό όπως στο Παράδειγμα 1. Παράδειγμα είναι η χρήση ESP με τη λειτουργία σήραγγας το οποίο μεταφέρει δεδομενογράμματα που χρησιμοποιούν AH με τη λειτουργία μεταφοράς. Η χρήση του AH από σταθμό-σε-σταθμό εξασφαλίζει την αυθεντικοποίηση και ακεραιότητα των δεδομένων από άκρο-σε-άκρο, ενώ η χρήση του ESP παρέχει εμπιστευτικότητα στα δεδομένα όσο αυτά διακινούνται στο δημόσιο και μη ασφαλές δίκτυο. Εάν η από σταθμό-σε-σταθμό σύνδεση χρησιμοποιεί επίσης το ESP, εξασφαλίζεται η εμπιστευτικότητα των δεδομένων και κατά τη διακίνησή τους εντός των δύο δικτύων.



Σχήμα 10.11: IPsec μεταξύ δύο πυλών.

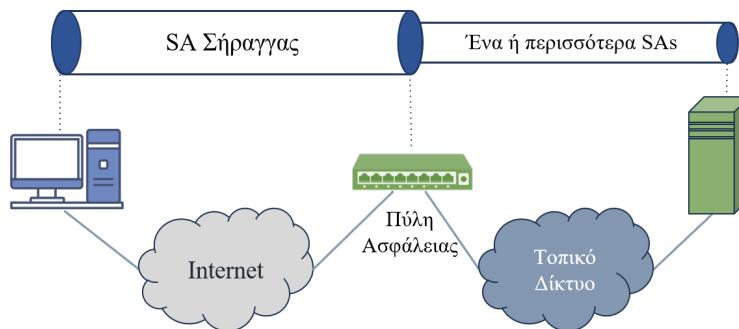


Σχήμα 10.12: IPsec μεταξύ δύο πυλών και σταθμών.

10.3.4.4 Παράδειγμα 4: Υποστήριξη Απομακρυσμένου Σταθμού

Έστω ότι θέλουμε να προστατεύσουμε την επικοινωνία ενός απομακρυσμένου σταθμού με το δίκτυο ενός οργανισμού στο οποίο υπάρχει μια πύλη (Σχήμα 10.13), όπως τυπικά ένα firewall. Ο απομακρυσμένος σταθμός μπορεί να χρησιμοποιεί σήραγγα για να προστατεύει την επικοινωνία με την πύλη σε ESP, εξασφαλίζοντας με αυτόν τον τρόπο την εμπιστευτικότητα στα δεδομένα που διακινούνται στο δημόσιο και μη ασφαλές δίκτυο.

Επιπλέον αυτής της προστασίας, η κίνηση από τον απομακρυσμένο σταθμό προς συγκεκριμένους κόμβους εντός του δικτύου, όπως είναι ένας εξυπηρετητής, μπορεί να προστατεύεται από μια εσωτερική σήραγγα, όπως και στο Παράδειγμα 1.



Σχήμα 10.13: Υποστήριξη απομακρυσμένου σταθμού.

10.4 Secure Shell Protocol

Το Secure Shell (SSH) σχεδιάστηκε αρχικά για να αντικαταστήσει μη ασφαλή πρωτόκολλα απομακρυσμένου κελύφους (shell) όπως το telnet. Έχει γίνει πλέον μια λύση γενικότερου σκοπού που χρησιμοποιείται για την παροχή ασφαλούς καναλιού επικοινωνίας μεταξύ δύο απομακρυσμένων κόμβων, καθώς και για εφαρμογές

όπως η ασφαλής απομακρυσμένη πρόσβαση σε πόρους, η απομακρυσμένη εκτέλεση εντολών, η παράδοση ενημερώσεων κώδικα λογισμικού και άλλες διαχειριστικές εργασίες.

Το SSH αναπτύχθηκε αρχικά από τον Tatu Ylonen το 1995 ως απάντηση σε ένα περιστατικό παραβίασης ασφαλείας στο Φινλανδικό πανεπιστημιακό δίκτυο. Ένα λογισμικό καταγραφής κωδικών πρόσβασης (keylogger) είχε εγκατασταθεί σε έναν εξυπηρετητή και όταν ανακαλύφθηκε είχε χιλιάδες ονόματα χρηστών και κωδικούς πρόσβασης στη βάση δεδομένων του. Αυτό το περιστατικό ώθησε τον Ylonen να αναπτύξει μια λύση που θα μπορούσε να χρησιμοποιήσει ο ίδιος για απομακρυσμένη σύνδεση μέσω διαδικτύου με ασφάλεια. Σήμερα, το πρωτόκολλο χρησιμοποιείται για τη διαμόρφωση, τη διαχείριση, τη συντήρηση και τη λειτουργία τειχών προστασίας, δρομολογητών, μεταγωγέων και διακομιστών. Είναι επίσης ενσωματωμένο σε πολλές λύσεις μεταφοράς αρχείων και διαχείρισης συστημάτων.

Το SSHv2 τυποποιήθηκε σε μια συλλογή RFCs το 2006 [14, 15, 16, 17, 18]. Η αρχική έκδοση, το SSHv1 έχει αρκετές σχεδιαστικές αδυναμίες και δεν πρέπει πλέον να χρησιμοποιείται. Το OpenSSH [19] είναι η πιο ευρέως χρησιμοποιούμενη υλοποίηση του πρωτοκόλλου.

10.4.1 Τρόπος Λειτουργίας του SSH

Το SSH χρησιμοποιεί το μοντέλο πελάτη-διακομιστή. Για τη δημιουργία ενός ασφαλούς καναλιού SSH, τα δύο μέρη θα πρέπει στο πλαίσιο μιας TCP σύνδεσης, να διαπραγματευτούν τον αριθμό έκδοσης και τους αλγόριθμους που θα χρησιμοποιηθούν και να δημιουργήσουν τα κοινόχρηστα κλειδιά συνόδου ώστε να μπορεί να ακολουθήσει μια ανταλλαγή δεδομένων προστατευμένη με συμμετρική κρυπτογράφηση. Αφού ολοκληρωθεί η αυθεντικοποίηση χρήστη, τα δύο μέρη μπορούν να δημιουργήσουν μια σύνοδο για την ανταλλαγή δεδομένων.

Έτσι, υπάρχουν δύο στάδια για τη δημιουργία μιας σύνδεσης: στο πρώτο στάδιο, και τα δύο συστήματα πρέπει να συμφωνήσουν στους αλγορίθμους κρυπτογράφησης για την προστασία των μηνυμάτων που θα ακολουθήσουν ενώ στο δεύτερο, ο χρήστης πρέπει να αυθεντικοποιηθεί από το σύστημα. Το επίπεδο μεταφοράς του SSH είναι υπεύθυνο για την αρχική ανταλλαγή και εγκαθίδρυση κλειδιών, την αυθεντικοποίηση διακομιστή και την εμπιστευτικότητα και την ακεραιότητα των μηνυμάτων που αποστέλλονται μέσω του ασφαλούς καναλιού. Στις ακόλουθες ενότητες περιγράφονται οι φάσεις λειτουργίας του SSH, όπως απεικονίζονται στο Σχήμα 10.14.

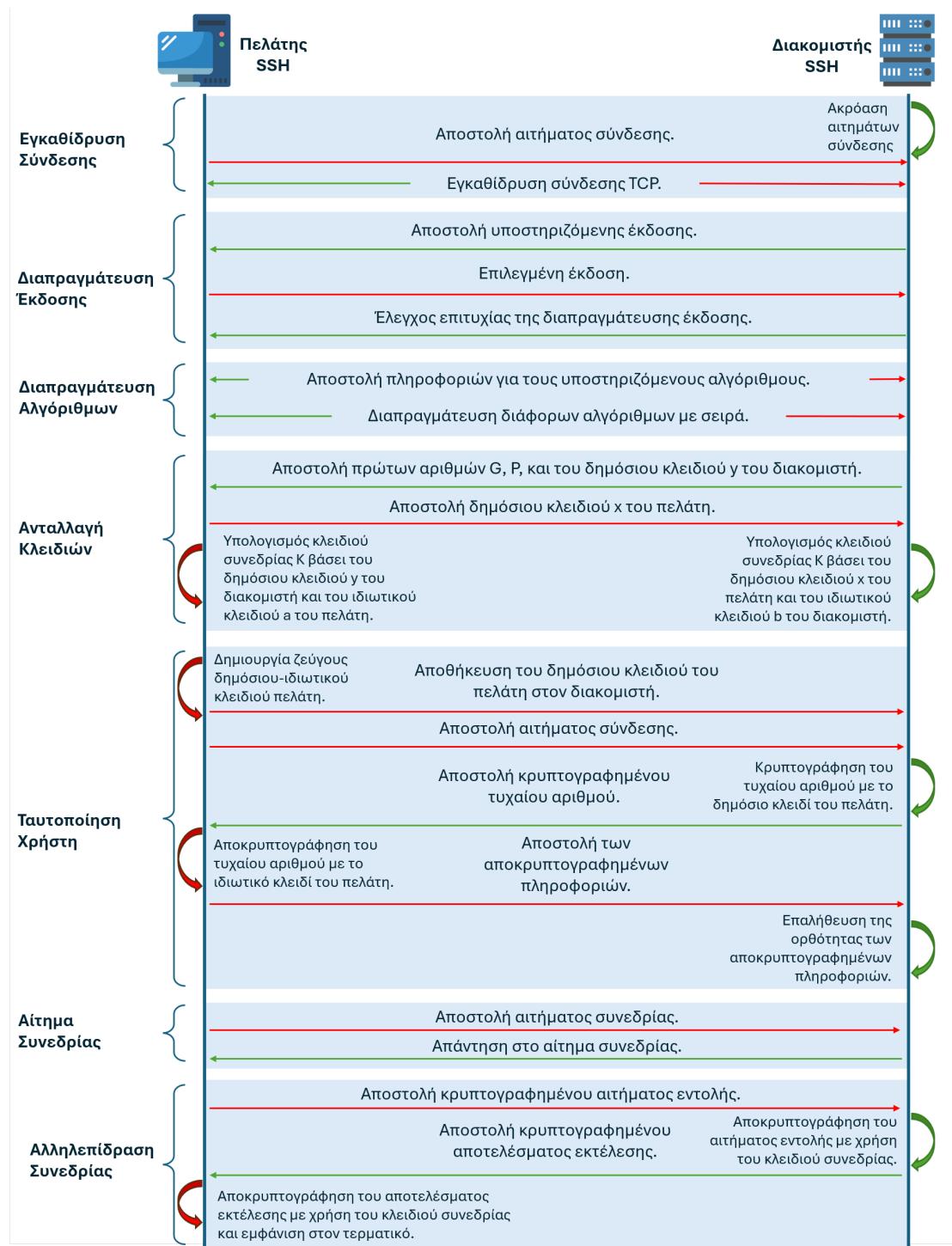
10.4.1.1 Διαπραγμάτευση Έκδοσης

Το SSH έχει δύο εκδόσεις: SSH1.X και SSH2.0. Σε σύγκριση με το SSH1.X, το SSH2.0 υποστηρίζει περισσότερες μεθόδους αυθεντικοποίησης και ανταλλαγής κλειδιών. Η διαδικασία διαπραγμάτευσης της έκδοσης μεταξύ του διακομιστή SSH και του πελάτη περιλαμβάνει τα ακόλουθα βήματα:

1. Ο διακομιστής SSH στέλνει τις υποστηριζόμενες πληροφορίες έκδοσης SSH στον πελάτη SSH.
2. Αφού λάβει τις πληροφορίες έκδοσης, ο πελάτης SSH καθορίζει την έκδοση που θα χρησιμοποιηθεί με βάση την έκδοση SSH που υποστηρίζει και στέλνει την επιλογή του στον διακομιστή SSH.
3. Ο διακομιστής SSH ελέγχει αν υποστηρίζει την έκδοση που πρότεινε ο πελάτης και, εάν ναι, η διαπραγμάτευση έκδοσης ολοκληρώνεται με επιτυχία.

10.4.1.2 Διαπραγμάτευση Αλγορίθμου

Το SSH χρησιμοποιεί πολλούς τύπους κρυπτογραφικών αλγορίθμων και παραμέτρων, συμπεριλαμβανομένου του αλγορίθμου ανταλλαγής κλειδιών για τη δημιουργία κλειδιών συνόδου, του αλγορίθμου συμμετρικής κρυπτογράφησης για την κρυπτογράφηση δεδομένων, και του αλγορίθμου δημοσίου κλειδιού για την αυθεντικοποίηση. Ο διακομιστής SSH και ο πελάτης υποστηρίζουν πολλαπλούς αλγορίθμους κάθε τύπου



Σχήμα 10.14: Οι φάσεις λειτουργίας του SSH.

και επομένως πρέπει να διαπραγματευτούν και να καθορίσουν τον αλγόριθμο που θα χρησιμοποιηθεί σε κάθε τύπο. Η διαδικασία έχει ως εξής:

1. Ο διακομιστής SSH και ο πελάτης ανταλλάσσουν τους υποστηριζόμενους αλγορίθμους.
2. Ο διακομιστής SSH και ο πελάτης διαπραγματεύονται τον αλγόριθμο που θα χρησιμοποιηθεί σε κάθε τύπο. Κατά τη διαπραγμάτευση κάθε τύπου αλγορίθμου, ο διακομιστής SSH και ο πελάτης επιλέγουν τους βέλτιστους αλγορίθμους που υποστηρίζουν και οι δύο. Εάν δεν μπορούν να βρεθούν κοινόχρηστα

αποδεκτοί αλγόριθμοι για κάποιον τύπο, η διαπραγμάτευση αλγορίθμου αυτού του τύπου αποτυγχάνει και η σύνδεση SSH τερματίζεται.

10.4.1.3 Ανταλλαγή Κλειδιών

Ο διακομιστής SSH και ο πελάτης χρησιμοποιούν έναν αλγόριθμο ανταλλαγής κλειδιών για να δημιουργήσουν δυναμικά ένα κοινόχρηστο κλειδί συνόδου και ένα αναγνωριστικό συνόδου (session ID) που χρησιμοποιείται για την αναγνώριση της σχετικής σύνδεσης SSH κατά την αυθεντικοποίηση. Οι υποστηριζόμενοι αλγόριθμοι ανταλλαγής κλειδιών στο SSH περιλαμβάνουν το Diffie-Hellman και την κρυπτογραφία ελλειπτικών καμπυλών. Μόλις καθοριστούν τα κλειδιά, όλα τα μηνύματα αποστέλλονται κρυπτογραφημένα μέσω του καναλιού χρησιμοποιώντας το Binary Packet Protocol (BPP) [16]. Αυτό καθορίζει ένα σχήμα κρυπτογράφησης που βασίζεται σε μια κατασκευή Encode-then-Encrypt-and-MAC (Κωδικοποίηση-Κρυπτογράφηση-και-MAC).

10.4.1.4 Αυθεντικοποίηση Πελάτη

Το SSH υποστηρίζει τους ακόλουθους μηχανισμούς αυθεντικοποίησης του πελάτη:

- **Με τη χρήση κωδικού πρόσβασης:** Ο πελάτης διαθέτει κωδικό πρόσβασης που του έχει διατεθεί από το σύστημα και τον οποίο στέλνει κρυπτογραφημένο, μαζί με το όνομα χρήστη, κάνοντας χρήση του δημοσίου κλειδιού του διακομιστή SSH.
- **Με τη χρήση δημοσίου κλειδιού:** Ο πελάτης διαθέτει ένα ζεύγος ιδιωτικού-δημοσίου κλειδιού το οποίο και χρησιμοποιεί για την αυθεντικοποίηση. Η διαδικασία αποτελείται από τα ακόλουθα βήματα:
 - Ο πελάτης ο οποίος προσπαθεί να αποκτήσει πρόσβαση στους πόρους του διακομιστή SSH ξεκινά μια σύνδεση με αυτόν.
 - Ο διακομιστής SSH χρησιμοποιεί το δημόσιο κλειδί του πελάτη για να κρυπτογραφήσει μια πρόκληση η οποία αποτελείται τυπικά από έναν τυχαίο αριθμό.
 - Ο διακομιστής SSH στέλνει ένα αίτημα προς τον πελάτη για να υπογράψει την κρυπτογραφημένη πρόκληση χρησιμοποιώντας το ιδιωτικό του κλειδί.
 - Ο πελάτης αποκρυπτογραφεί την πρόκληση χρησιμοποιώντας το ιδιωτικό του κλειδί και την επιστρέφει στον διακομιστή.
 - Ο διακομιστής ελέγχει αν η αποκρυπτογραφημένη τιμή είναι η σωστή. Αν ναι, η διαδικασία αυθεντικοποίησης ολοκληρώνεται επιτυχώς.

Μια πλήρης λίστα με κρυπτογραφικές συνίτες για το SSH παρατίθεται από τον οργανισμό [Internet Assigned Numbers Authority \(IANA\)](#). Ανάμεσα τους ξεχωρίζουν οι ακόλουθες:

- AES-128_CTR με HMAC_SHA2-256 ή HMAC_SHA2-512
- AES-192_CTR με HMAC_SHA2-256 ή HMAC_SHA2-512
- AES-256_CTR με HMAC_SHA2-256 ή HMAC_SHA2-512
- AEAD_AES-128_GCM
- AEAD_AES-256_GCM

Βιβλιογραφία

- [1] T. Dierks and C. Allen. *The TLS Protocol Version 1.0.* en. Tech. rep. RFC2246. RFC Editor, Jan. 1999, RFC2246. doi: 10.17487/rfc2246. URL: <https://www.rfc-editor.org/info/rfc2246> (visited on 11/23/2022).
- [2] T. Dierks and E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.1.* en. Tech. rep. RFC4346. RFC Editor, Apr. 2006, RFC4346. doi: 10.17487/rfc4346. URL: <https://www.rfc-editor.org/info/rfc4346> (visited on 11/23/2022).
- [3] T. Dierks and E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.2.* en. Tech. rep. RFC5246. RFC Editor, Aug. 2008, RFC5246. doi: 10.17487/rfc5246. URL: <https://www.rfc-editor.org/info/rfc5246> (visited on 11/23/2022).
- [4] E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.3.* en. Tech. rep. RFC8446. RFC Editor, Aug. 2018, RFC8446. doi: 10.17487/rfc8446. URL: <https://www.rfc-editor.org/info/rfc8446> (visited on 10/29/2022).
- [5] H. Krawczyk and P. Eronen. *HMAC-based Extract-and-Expand Key Derivation Function (HKDF).* en. Tech. rep. RFC5869. RFC Editor, May 2010, RFC5869. doi: 10.17487/rfc5869. URL: <https://www.rfc-editor.org/info/rfc5869> (visited on 11/24/2022).
- [6] S. Santesson et al. *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP.* en. Tech. rep. RFC6960. RFC Editor, June 2013, RFC6960. doi: 10.17487/rfc6960.
- [7] Karen Seo and Stephen Kent. *Security Architecture for the Internet Protocol.* RFC 4301. Dec. 2005. doi: 10.17487/RFC4301. URL: <https://www.rfc-editor.org/info/rfc4301>.
- [8] Stephen Kent. *IP Authentication Header.* RFC 4302. Dec. 2005. doi: 10.17487/RFC4302. URL: <https://www.rfc-editor.org/info/rfc4302>.
- [9] Stephen Kent. *IP Encapsulating Security Payload (ESP).* RFC 4303. Dec. 2005. doi: 10.17487/RFC4303. URL: <https://www.rfc-editor.org/info/rfc4303>.
- [10] Paul Wouters, Daniel Migault, John Preuß Mattsson, Yoav Nir, and Tero Kivinen. *Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH).* RFC 8221. Oct. 2017. doi: 10.17487/RFC8221. URL: <https://www.rfc-editor.org/info/rfc8221>.
- [11] Charlie Kaufman, Paul E. Hoffman, Yoav Nir, Pasi Eronen, and Tero Kivinen. *Internet Key Exchange Protocol Version 2 (IKEv2).* RFC 7296. Oct. 2014. doi: 10.17487/RFC7296. URL: <https://www.rfc-editor.org/info/rfc7296>.
- [12] Yoav Nir, Tero Kivinen, Paul Wouters, and Daniel Migault. *Algorithm Implementation Requirements and Usage Guidance for the Internet Key Exchange Protocol Version 2 (IKEv2).* RFC 8247. Sept. 2017. doi: 10.17487/RFC8247. URL: <https://www.rfc-editor.org/info/rfc8247>.
- [13] Dr. Steve E. Deering and Bob Hinden. *Internet Protocol, Version 6 (IPv6) Specification.* RFC 8200. July 2017. doi: 10.17487/RFC8200. URL: <https://www.rfc-editor.org/info/rfc8200>.
- [14] Chris M. Lonnqvist and Tatu Ylonen. *The Secure Shell (SSH) Protocol Architecture.* RFC 4251. Jan. 2006. doi: 10.17487/RFC4251. URL: <https://www.rfc-editor.org/info/rfc4251>.
- [15] Chris M. Lonnqvist and Tatu Ylonen. *The Secure Shell (SSH) Authentication Protocol.* RFC 4252. Jan. 2006. doi: 10.17487/RFC4252. URL: <https://www.rfc-editor.org/info/rfc4252>.
- [16] Chris M. Lonnqvist and Tatu Ylonen. *The Secure Shell (SSH) Transport Layer Protocol.* RFC 4253. Jan. 2006. doi: 10.17487/RFC4253. URL: <https://www.rfc-editor.org/info/rfc4253>.

- [17] D. Bider. *Extension Negotiation in the Secure Shell (SSH) Protocol*. en. Tech. rep. RFC8308. RFC Editor, Mar. 2018, RFC8308. doi: 10.17487/RFC8308. URL: <https://www.rfc-editor.org/info/rfc8308> (visited on 12/16/2022).
- [18] D. Bider. *Use of RSA Keys with SHA-256 and SHA-512 in the Secure Shell (SSH) Protocol*. en. Tech. rep. RFC8332. RFC Editor, Mar. 2018, RFC8332. doi: 10.17487/RFC8332. URL: <https://www.rfc-editor.org/info/rfc8332> (visited on 12/16/2022).
- [19] Open SSH Project. URL: [http://www.openssh.org/.](http://www.openssh.org/)

ΚΕΦΑΛΑΙΟ 11

ΑΛΥΣΙΔΕΣ ΜΠΛΟΚ

Περίληψη

Οι αλυσίδες μπλοκ αποτελούν διαφανή και ανθεκτικά σε παραβιάσεις, ψηφιακά καθολικά, που υλοποιούνται με κατανεμημένο τρόπο και συνήθως χωρίς την ύπαρξη κάποιας κεντρικής αρχής. Επιτρέπουν σε μια κοινότητα χρηστών να καταγράφουν συναλλαγές σε ένα κοινόχρηστο καθολικό εντός αυτής της κοινότητας, με τρόπο ώστε καμία συναλλαγή να μη μπορεί να μεταβληθεί μετά τη δημοσίευσή της στο δίκτυο της αλυσίδας μπλοκ. Έτσι, οι αλυσίδες μπλοκ επιτρέπουν σε ομότιμους κόμβους, που δεν έχουν σχέσεις εμπιστοσύνης μεταξύ τους, να ανταλλάσσουν ψηφιακά δεδομένα με λιγότερα ή καθόλου τρίτα μέρη ή μεσάζοντες. Τα δεδομένα θα μπορούσαν να αντιστοιχούν σε χρήματα, συμβόλαια, τίτλους γης, ιατρικά και εκπαιδευτικά αρχεία, πιστοποιητικά, αγορά και πώληση αγαθών και υπηρεσιών, ή οποιαδήποτε συναλλαγή ή περιουσιακό στοιχείο που μπορεί να ψηφιοποιηθεί. Στο κεφάλαιο αυτό περιγράφονται οι βασικές έννοιες της τεχνολογίας κατανεμημένου καθολικού και της τεχνολογίας αλυσίδας μπλοκ (Ενότητα 11.1) και αναλύονται τα χαρακτηριστικά των βασικών κατηγοριών αλυσίδων μπλοκ: αυτές με άδεια και χωρίς άδεια (Ενότητα 11.2). Στην Ενότητα 11.4 εξετάζονται οι αλγόριθμοι συναίνεσης που εξασφαλίζουν τη συμφωνία μεταξύ των συμμετεχόντων στο δίκτυο. Στη συνέχεια, η Ενότητα 11.5 εστιάζει σε δημοφιλείς πλατφόρμες αλυσίδων μπλοκ, όπως το Bitcoin (Ενότητα 11.5.1), το Ethereum (Ενότητα 11.5.2) και το Hyperledger Fabric (Ενότητα 11.5.3). Τέλος, η Ενότητα 11.6 αναλύει τις προκλήσεις ασφάλειας και λειτουργίας των αλυσίδων μπλοκ, όπως οι επιθέσεις Sybil, η επίθεση του 51% και η διπλή δαπάνη.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών εννοιών της Κρυπτογραφίας Δημοσίου Κλειδιού (Κεφάλαιο 2) και των Συναρτήσεων Σύνοψης (Κεφάλαιο 3).

11.1 Βασικές Έννοιες

11.1.1 Τεχνολογία Κατανεμημένου Καθολικού

Η Τεχνολογία Κατανεμημένου Καθολικού (Distributed Ledger Technology – DLT) είναι μια καινοτόμος προσέγγιση στην αποθήκευση και διαχείριση δεδομένων που εξαλείφει την ανάγκη για μια κεντρική αρχή. Σε ένα κατανεμημένο καθολικό, τα δεδομένα αποθηκεύονται σε πολλούς κόμβους του δικτύου. Κάθε κόμβος διατηρεί μια ακριβή και ενημερωμένη αντιγραφή του καθολικού, διασφαλίζοντας έτσι τη συνέπεια και την ακεραιότητα των δεδομένων.

Η DLT παρέχει αυξημένη ασφάλεια και ανθεκτικότητα, διότι για να παραβιαστεί το σύστημα, πρέπει να καταβληθεί μια συντονισμένη επίθεση σε πολλούς κόμβους ταυτόχρονα. Ένα χαρακτηριστικό παράδειγμα χρήσης της DLT είναι τα συστήματα πληρωμών, όπου οι συναλλαγές καταγράφονται και επαληθεύονται από πολλούς κόμβους, αποτρέποντας την παραποίηση των δεδομένων.

Η εξέλιξη της DLT οδήγησε στη δημιουργία διαφόρων εφαρμογών, όπως τα συστήματα ταυτοποίησης, όπου οι ταυτότητες των χρηστών αποθηκεύονται με ασφάλεια σε ένα κατανεμημένο καθολικό. Επιπλέον, οι αλυσίδες εφοδιασμού χρησιμοποιούν την DLT για να παρακολουθούν και να διασφαλίζουν την αυθεντικότητα και την ακεραιότητα των προϊόντων από την παραγωγή μέχρι την παράδοση.

11.1.2 Τεχνολογία Αλυσίδας Μπλοκ

Η Τεχνολογία Αλυσίδας Μπλοκ (Blockchain Technology) είναι ένας ειδικός τύπος DLT όπου τα δεδομένα οργανώνονται σε μπλοκ που συνδέονται μεταξύ τους με κρυπτογραφικές συνόψεις (hashes). Η IEEE ορίζει την αλυσίδα μπλοκ ως μια «κατανεμημένη βάση δεδομένων ανοιχτής πηγής, που χρησιμοποιεί κρυπτογραφία αιχμής μέσω κατανεμημένου καταλόγου καθολικού που επιτρέπει την εμπιστοσύνη μεταξύ διαφορετικών ατόμων ή τρίτων». Αυτή η αλυσίδα από μπλοκ εξασφαλίζει τη διαφάνεια και την ακεραιότητα των δεδομένων, καθώς κάθε νέο μπλοκ περιέχει τη σύνοψη του προηγούμενου μπλοκ, δημιουργώντας έτσι μια αδιάσπαστη αλυσίδα.

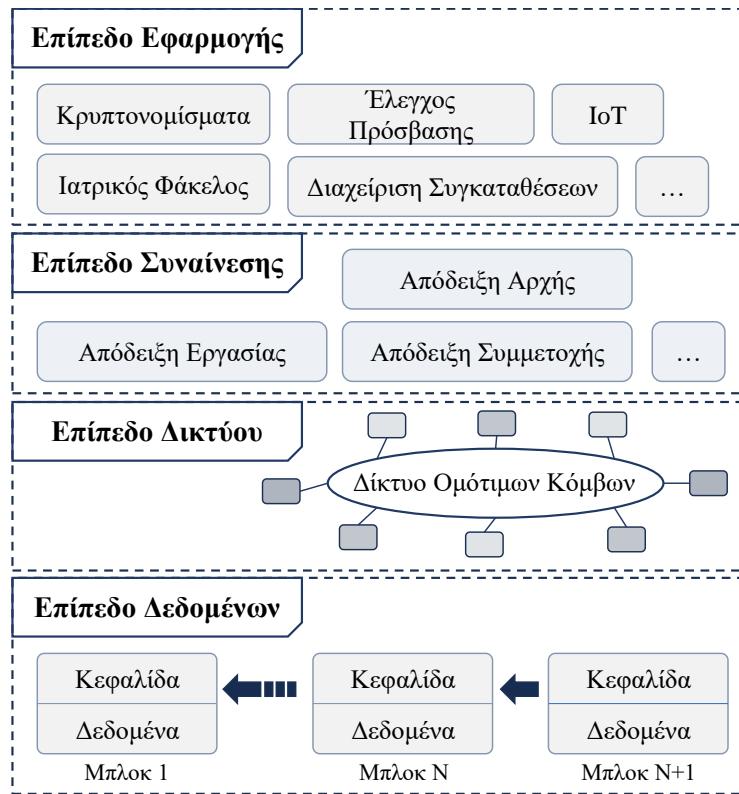
Η τεχνολογία αλυσίδας μπλοκ πρωτοεμφανίστηκε το 2008 με τη δημοσίευση του whitepaper του Satoshi Nakamoto για το Bitcoin [1]. Από τότε, η τεχνολογία αυτή έχει εξελιχθεί και υιοθετηθεί σε πολλούς τομείς πέραν των κρυπτονομισμάτων. Η δυνατότητα καταγραφής των συναλλαγών με διαφάνεια και ακεραιότητα, χωρίς την ανάγκη κεντρικής αρχής, έχει καταστήσει την αλυσίδα μπλοκ ιδιαίτερα δημοφιλή σε διάφορους τομείς.

Η δομή μιας αλυσίδας μπλοκ επιτρέπει την ανίχνευση οποιασδήποτε αλλοίωσης στα δεδομένα, καθώς κάθε αλλαγή σε ένα μπλοκ θα απαιτούσε την τροποποίηση όλων των επόμενων μπλοκ, κάτι που είναι πρακτικά αδύνατο σε ένα κατανεμημένο δίκτυο με πολλούς κόμβους.

Το Σχήμα 11.1 παρουσιάζει την πολυεπίπεδη δομή που χρησιμοποιείται στις υποδομές αλυσίδων μπλοκ. Στην κορυφή βρίσκεται το επίπεδο εφαρμογής, όπου οι εφαρμογές επιτρέπουν στους τελικούς χρήστες να αλληλεπιδρούν με την αλυσίδα μπλοκ. Το επόμενο επίπεδο, το επίπεδο συναίνεσης, αποτελεί την καρδιά του συστήματος των αλυσίδων μπλοκ. Ο αλγόριθμος συναίνεσης παίζει κρίσιμο ρόλο στην επικύρωση και την οργάνωση των μπλοκ, καθώς και στη διασφάλιση της ομοφωνίας μεταξύ όλων των συμμετεχόντων σχετικά με την τρέχουσα κατάσταση της αλυσίδας μπλοκ. Το επίπεδο δίκτυου φροντίζει για την επικοινωνία μεταξύ των ομότιμων κόμβων και την ανταλλαγή συναλλαγών που περιέχονται σε μπλοκ. Τέλος, το επίπεδο δεδομένων ασχολείται με ένα σύνολο συνδεδεμένων μπλοκ, στα οποία οι συναλλαγές είναι ταξινομημένες.

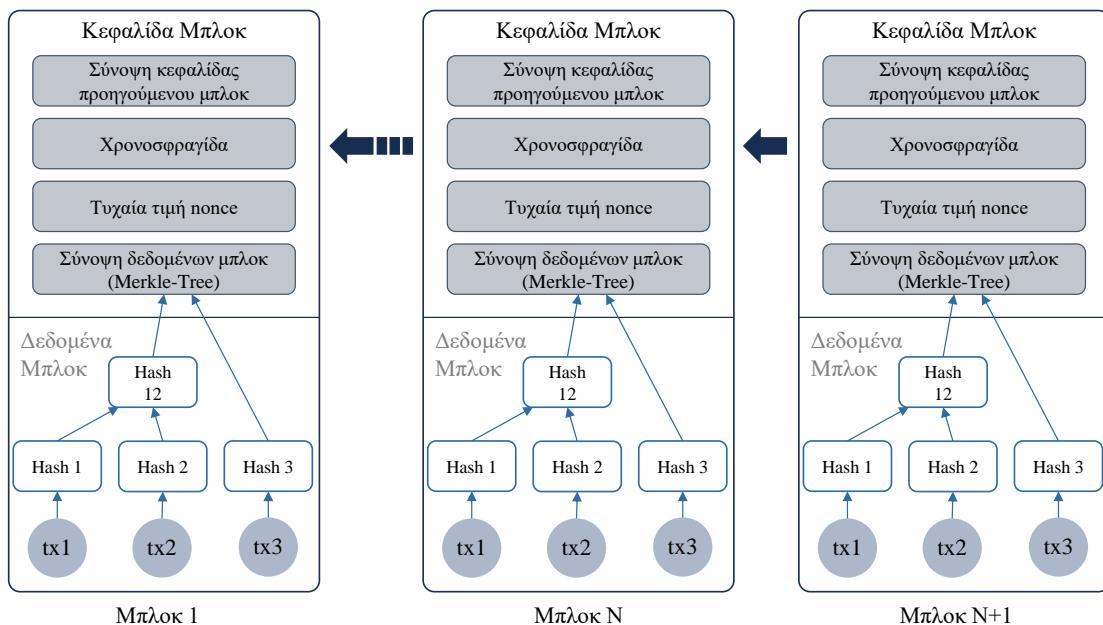
11.1.3 Μπλοκ

Ένα μπλοκ (block) είναι η βασική μονάδα αποθήκευσης δεδομένων σε μια αλυσίδα μπλοκ. Κάθε μπλοκ περιέχει έναν αριθμό συναλλαγών που έχουν επαληθευτεί από τους κόμβους του δικτύου. Τα βασικά στοιχεία ενός μπλοκ περιλαμβάνουν τη σύνοψη του προηγούμενου μπλοκ, η οποία διασφαλίζει τη σύνδεση του μπλοκ με το προηγούμενο και δημιουργεί μια συνεχή αλυσίδα. Επιπλέον, περιλαμβάνει μια χρονική σφραγίδα (timestamp) που προσδιορίζει την ώρα δημιουργίας του μπλοκ, και αναλόγως με τον αλγόριθμο συναίνεσης, ένα



Σχήμα 11.1: Πολυεπίπεδη αρχιτεκτονική αλυσίδας μπλοκ.

nonce, τον τυχαίο αριθμό που χρησιμοποιείται στη διαδικασία εξόρυξης για την επίλυση του προβλήματος συναίνεσης. Οι πληροφορίες αυτές αποθηκεύονται στην κεφαλίδα του μπλοκ ενώ στο σώμα του μπλοκ αποθηκεύεται μια λίστα συναλλαγών (tx), όπως απεικονίζεται στο Σχήμα 11.2.



Σχήμα 11.2: Δομή των μπλοκ σε μια αλυσίδα μπλοκ.

Για παράδειγμα, ένα τυπικό μπλοκ στο δίκτυο Bitcoin περιέχει τη σύνοψη του προηγούμενου μπλοκ, τη

σύνοψη των συναλλαγών του και το nonce που χρησιμοποιείται για την επαλήθευση του μπλοκ μέσω της διαδικασίας εξόρυξης (mining). Η διαδικασία εξόρυξης περιλαμβάνει την επίλυση ενός περίπλοκου μαθηματικού προβλήματος, και ο πρώτος κόμβος που θα βρει τη λύση προσθέτει ένα νέο μπλοκ στην αλυσίδα και λαμβάνει μια ανταμοιβή, όπως θα αναλυθεί παρακάτω.

11.1.4 Συναλλαγές

Μια συναλλαγή (transaction) αποτυπώνει τη μεταφορά δεδομένων ή αξίας μεταξύ δύο ή περισσότερων οντοτήτων. Οι συναλλαγές περιλαμβάνουν διάφορα στοιχεία, όπως τη διεύθυνση του αποστολέα, τη διεύθυνση του παραλήπτη, το ποσό της αξίας που μεταφέρεται ή τα δεδομένα που καταγράφονται στην αλυσίδα μπλοκ.

Η διαδικασία μιας συναλλαγής σε ένα δίκτυο αλυσίδας μπλοκ περιλαμβάνει διάφορα στάδια. Αρχικά, ο αποστολέας δημιουργεί τη συναλλαγή με τα απαιτούμενα στοιχεία και την υπογράφει με το ιδιωτικό του κλειδί, εξασφαλίζοντας έτσι την ακεραιότητα και την αυθεντικότητα της συναλλαγής. Στη συνέχεια, η υπογεγραμμένη συναλλαγή μεταδίδεται στους κόμβους του δικτύου, όπου επαληθεύεται η εγκυρότητά της με το δημόσιο κλειδί του υπογράφοντος, και γίνονται και οι απαραίτητοι πρόσθετοι έλεγχοι όπως θα αναλυθούν παρακάτω (π.χ. το ενδεχόμενο Διπλής Δαπάνης – Ενότητα 11.6.6). Η έγκυρη συναλλαγή θα οριστικοποιηθεί με την προσθήκη της ως μέρος ενός νέου μπλοκ στην αλυσίδα.

Για παράδειγμα, σε ένα δίκτυο όπως το Bitcoin, μια τυπική συναλλαγή μπορεί να περιλαμβάνει τη μεταφορά 0.5 bitcoin (BTC) από τον χρήστη A στον χρήστη B. Η συναλλαγή αυτή θα περιλαμβάνει τις δημόσιες διευθύνσεις του A και του B, το ποσό των 0.5 BTC, και την υπογραφή του A στα δεδομένα της συναλλαγής. Η επαλήθευση της συναλλαγής εξασφαλίζει ότι ο A διαθέτει τα απαιτούμενα BTC και ότι η συναλλαγή δεν έχει αλλοιωθεί.

Η καταγραφή των συναλλαγών στην αλυσίδα μπλοκ διασφαλίζει την ακεραιότητα και τη διαφάνεια των δεδομένων, καθιστώντας πρακτικά αδύνατη την αλλοίωση ή την παραποίηση των καταγεγραμμένων συναλλαγών. Αυτό έχει σημαντικές επιπτώσεις σε διάφορους τομείς, όπως οι χρηματοοικονομικές υπηρεσίες, όπου η αξιοπιστία και η διαφάνεια είναι ζωτικής σημασίας.

Το πρώτο μπλοκ μιας αλυσίδας μπλοκ ονομάζεται genesis block και αποτελεί τη βάση πάνω στην οποία χτίζεται ολόκληρη η αλυσίδα. Αυτό το μπλοκ έχει την ιδιαιτερότητα πως δεν έχει προηγούμενο μπλοκ να συνδεθεί με αυτό. Ο ρόλος του είναι θεμελιώδης, διότι καθορίζει την αρχή της αλυσίδας και όλα τα επόμενα μπλοκ προστίθενται και επαληθεύονται με βάση αυτό το αρχικό μπλοκ. Στην περίπτωση του Bitcoin, το genesis block δημιουργήθηκε από τον Satoshi Nakamoto στις 3 Ιανουαρίου 2009. Σε αντίθεση με οποιοδήποτε από τα εκατοντάδες χιλιάδες μπλοκ που ακολούθησαν, ο Nakamoto άφησε ένα μήνυμα στον κωδικό του μπλοκ: "The Times 03/Jan/2009 Chancellor on brink of second bailout for banks". Αυτή η πρόταση προέρχεται κατευθείαν από τον τίτλο ενός άρθρου των London Times με ημερομηνία 3 Ιανουαρίου 2009, το οποίο περιγράφει λεπτομερώς τις τράπεζες που διασώθηκαν από τη βρετανική κυβέρνηση. Το hash του genesis block του Bitcoin είναι:

0000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f

και περιέχει μια ανταμοιβή εξόρυξης (mining reward) αξίας 50 BTC μόνο (δηλ., χωρίς συναλλαγές μεταφοράς ποσών). Το genesis block θέτει τα θεμέλια για την ασφάλεια και την ακεραιότητα της αλυσίδας μπλοκ, καθώς οποιαδήποτε αλλαγή σε αυτό θα είχε ως αποτέλεσμα την αλλοίωση όλων των επόμενων μπλοκ, καθιστώντας το σύστημα αναξιόπιστο.

11.2 Κατηγοριοποίηση των Αλυσίδων Μπλοκ

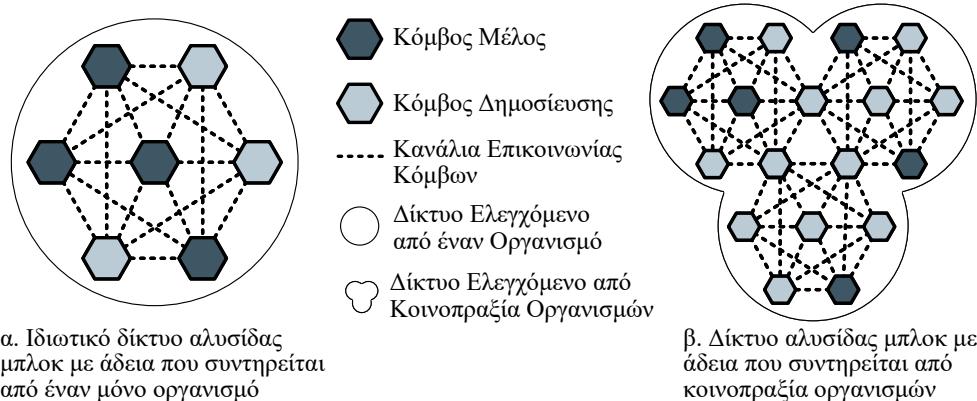
Η τεχνολογία αλυσίδας μπλοκ έχει εξελιχθεί σημαντικά από την εμφάνιση του Bitcoin το 2008. Αυτή η εξέλιξη έχει οδηγήσει στη δημιουργία διαφόρων τύπων αλυσίδων μπλοκ, κάθε μια με διαφορετικά χαρακτηριστικά και χρήσεις. Η κατηγοριοποίηση των αλυσίδων μπλοκ αποτελεί βασικό στοιχείο για την κατανόηση των δια-

φόρων αρχιτεκτονικών και εφαρμογών που χρησιμοποιούνται σήμερα. Σε γενικές γραμμές, οι αλυσίδες μπλοκ μπορούν να διαχωριστούν σε δύο κύριες κατηγορίες: αλυσίδες μπλοκ με άδεια και αλυσίδες μπλοκ χωρίς άδεια. Αυτή η κατηγοριοποίηση βασίζεται στα δικαιώματα πρόσβασης και στον τρόπο διαχείρισης του συστήματος. Κάθε κατηγορία έχει τα δικά της πλεονεκτήματα και μειονεκτήματα, ενώ η επιλογή της κατάλληλης κατηγορίας εξαρτάται από τις συγκεκριμένες ανάγκες και απαιτήσεις της εκάστοτε εφαρμογής.

11.2.1 Αλυσίδες Μπλοκ με Άδεια

Οι αλυσίδες μπλοκ με άδεια (permissioned blockchains) είναι δίκτυα όπου τόσο οι κόμβοι που δημιουργούν μπλοκ όσο και οι χρήστες πρέπει να είναι εξουσιοδοτημένοι από κάποια αρχή, είτε κεντρική είτε αποκεντρωμένη. Καθώς μόνο εξουσιοδοτημένοι κόμβοι και χρήστες συντηρούν την αλυσίδα μπλοκ, τόσο η ανάγνωση της αλυσίδας όσο και η εισαγωγή συναλλαγών σε αυτή, γίνονται ελεγχόμενα από την αρχή εξουσιοδότησης. Έτσι, στις αλυσίδες μπλοκ με άδεια μπορούν είτε να επιτρέπουν σε οποιονδήποτε να διαβάζει την αλυσίδα μπλοκ είτε να περιορίζουν την ανάγνωση μόνο σε εξουσιοδοτημένα άτομα ή κόμβους. Ομοίως, μπορούν να επιτρέπουν σε οποιονδήποτε να υποβάλει συναλλαγές ή να περιορίζουν αυτό το δικαίωμα μόνο σε εξουσιοδοτημένα άτομα ή κόμβους.

Σε μια αλυσίδα μπλοκ με άδεια (Σχήμα 11.3), οι δύο βασικότεροι ρόλοι είναι ο κόμβος επικύρωσης (validator node), ο οποίος είναι υπεύθυνος για την επικύρωση των συναλλαγών και τη δημιουργία νέων μπλοκ, εξασφαλίζοντας τη συμμόρφωση με τους κανόνες του δικτύου, και ο κόμβος μέλος (Member Node), ο οποίος συμμετέχει στο δίκτυο υποβάλλοντας συναλλαγές χωρίς δικαιώματα επικύρωσης ή δημιουργίας μπλοκ, ενώ διαθέτει περιορισμένη πρόσβαση σε δεδομένα. Εκτός από αυτούς, υπάρχουν και άλλοι, λιγότερο βασικοί ρόλοι, όπως οι κόμβοι ομότιμοι (Peer Nodes), που ανταλλάσσουν δεδομένα με άλλους κόμβους χωρίς να επικυρώνουν συναλλαγές, οι κόμβοι υποστήριξης (Endorser Nodes), που επαληθεύουν και υπογράφουν τις συναλλαγές πριν από την τελική τους επικύρωση, και οι διαχειριστικοί κόμβοι (Admin Nodes), που είναι υπεύθυνοι για τη διαχείριση του δικτύου και τα δικαιώματα των άλλων κόμβων, διασφαλίζοντας την εύρυθμη λειτουργία και ασφάλεια του συστήματος.



Σχήμα 11.3: Αναπαραστάσεις δικτύων αλυσίδας μπλοκ με άδεια.

Οι αλυσίδες μπλοκ με άδεια παρέχουν παρόμοιες λειτουργίες με τις αλυσίδες μπλοκ χωρίς άδεια, όπως η ιχνηλασμότητα των ψηφιακών αγαθών στην αλυσίδα μπλοκ. Χρησιμοποιούν επίσης αντίστοιχους αλγόριθμους συνάνεσης για τη δημιουργία μπλοκ, αλλά αυτοί οι αλγόριθμοι δεν απαιτούν την κατανάλωση υπερβολικών πόρων όπως συμβαίνει σε κάποιες αλυσίδες μπλοκ χωρίς άδεια, π.χ. στο Bitcoin. Ωστόσο, η ταυτοποίηση των κόμβων είναι απαραίτητη για τη συμμετοχή στη δημιουργία μπλοκ, και η εξουσιοδότησή τους μπορεί να ανακληθεί εάν δεν ακολουθούν σωστά τον αλγόριθμο συναίνεσης. Αυτοί οι αλγόριθμοι τυπικά επιτρέπουν στις αλυσίδες μπλοκ με άδεια να λειτουργούν πιο γρήγορα και να απαιτούν λιγότερους υπολογιστικούς πόρους.

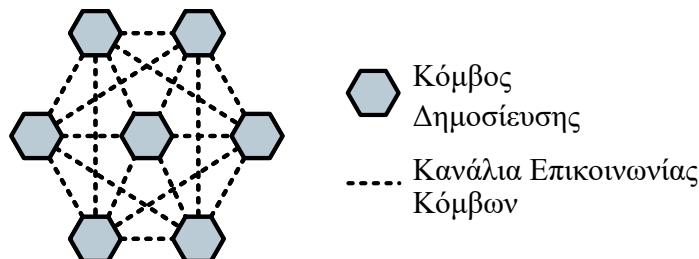
Οι αλυσίδες μπλοκ με άδεια χρησιμοποιούνται κυρίως από οργανισμούς που χρειάζονται αυστηρότερο

έλεγχο και προστασία των δεδομένων τους. Ωστόσο, αν μια μοναδική οντότητα ελέγχει τη συμμετοχή στη δημιουργία μπλοκ, οι χρήστες της αλυσίδας μπλοκ πρέπει να εμπιστεύονται πλήρως αυτή την οντότητα (Σχήμα 11.3.α). Τα δίκτυα αυτά μπορούν επίσης να χρησιμοποιηθούν από κοινοπράξies οργανισμών που θέλουν να συνεργαστούν αλλά δεν εμπιστεύονται απόλυτα ο ένας τον άλλον (Σχήμα 11.3.β). Σε αυτή την περίπτωση, δημιουργούν ένα δίκτυο αλυσίδας μπλοκ με άδεια και προσκαλούν άλλους συνεργαζόμενους οργανισμούς να καταγράφουν τις συναλλαγές τους σε ένα κοινό κατανεμημένο καθολικό. Αυτοί οι οργανισμοί μπορούν να καθορίσουν τον κατάλληλο αλγόριθμο συναίνεσης, ανάλογα με το επίπεδο εμπιστοσύνης μεταξύ τους. Πέρα από την εμπιστοσύνη, αυτά τα δίκτυα παρέχουν διαφάνεια, βοηθώντας στη λήψη καλύτερων επιχειρηματικών αποφάσεων και στην λογοδοσία των μερών που επιδεικνύουν κακόβουλη συμπεριφορά. Επιπλέον, μπορεί να περιλαμβάνουν εποπτικές οντότητες που ελέγχουν τις διαδικασίες λειτουργίας της αλυσίδας μπλοκ.

Οι αλυσίδες μπλοκ με άδεια χρησιμοποιούνται ευρέως στον επιχειρηματικό τομέα για την αυτοματοποίηση και την ασφάλεια των διαδικασιών. Για παράδειγμα, το Hyperledger Fabric είναι ένα από τα πιο γνωστά πλαίσια για αλυσίδες μπλοκ με άδεια, προσφέροντας ευελιξία και ασφάλεια για επιχειρηματικές εφαρμογές.

11.2.2 Αλυσίδες Μπλοκ χωρίς Άδεια

Οι αλυσίδες μπλοκ χωρίς άδεια (permissionless blockchains) λειτουργούν σε ανοικτά δίκτυα που επιτρέπουν σε οποιονδήποτε να συμμετέχει χωρίς να απαιτείται εξουσιοδότηση από κάποια αρχή (βλέπε Σχήμα 11.4). Είναι σχεδιασμένες να είναι ανθεκτικές σε λογοκρισία και επιθέσεις, καθώς η αποκεντρωμένη φύση τους καθιστά δύσκολη την παρέμβαση από έναν μεμονωμένο κόμβο ή ομάδα κόμβων. Αυτό τις καθιστά ιδιαίτερα κατάληγες για εφαρμογές όπου η εμπιστοσύνη δεν μπορεί να δοθεί σε μια κεντρική αρχή, όπως τα κρυπτονομίσματα.



Σχήμα 11.4: Αναπαράσταση ενός δημόσιου δικτύου αλυσίδας μπλοκ χωρίς άδεια.

Τα Bitcoin και Ethereum είναι δύο από τα πιο γνωστά παραδείγματα αλυσίδων μπλοκ χωρίς άδεια. Στα δίκτυα αυτά, οποιοσδήποτε μπορεί να γίνει κόμβος, να επαληθεύει συναλλαγές και να συμμετέχει στη διαδικασία δημιουργίας νέων μπλοκ. Αυτός ο ανοικτός χαρακτήρας των δικτύων αυτών προωθεί την καινοτομία και την ανάπτυξη νέων εφαρμογών.

11.2.3 Δικαιώματα και Διαφορές

Οι κύριες διαφορές μεταξύ αλυσίδων μπλοκ με άδεια και χωρίς άδεια σχετίζονται κυρίως με ζητήματα ελέγχου πρόσβασης. Στις αλυσίδες μπλοκ με άδεια, οι συμμετέχοντες πρέπει να λάβουν έγκριση για να ενταχθούν στο δίκτυο. Αυτό παρέχει υψηλότερα επίπεδα ασφάλειας και ελέγχου, καθώς η διαχείριση μπορεί να εξασφαλίσει ότι μόνο αξιόπιστοι κόμβοι συμμετέχουν. Αυτή η προσέγγιση είναι ιδανική για επιχειρηματικές εφαρμογές, όπου η ασφάλεια των δεδομένων είναι κρίσιμη.

Από την άλλη πλευρά, στις αλυσίδες μπλοκ χωρίς άδεια, μπορεί να συμμετέχει οποιαδήποτε οντότητα, γεγονός που προάγει την αποκέντρωση και την ανθεκτικότητα του δικτύου. Ωστόσο, η ανοιχτή φύση τους μπορεί να οδηγήσει σε προβλήματα ασφάλειας, όπως οι επιθέσεις Sybil (αναλύεται στην Ενότητα 11.6.1, όπου

ένας κακόβουλος χρήστης δημιουργεί πολλαπλές ψευδείς ταυτότητες για να αποκτήσει τον έλεγχο της πλειοψηφίας των κόμβων του δικτύου. Παρά τα πιθανά προβλήματα, η αποκεντρωμένη φύση τους καθιστά αυτές τις αλυσίδες μπλοκ ιδανικές για εφαρμογές όπου η διαφάνεια και η ανεξαρτησία είναι απαραίτητες.

11.3 Βασικά Χαρακτηριστικά των Αλυσίδων Μπλοκ

11.3.1 Συναρτήσεις Σύνοψης

Οι συναρτήσεις σύνοψης (hash functions) αποτελούν ένα από τα βασικά και πιο σημαντικά χαρακτηριστικά της τεχνολογίας αλυσίδας μπλοκ. Όπως αναλύθηκε και στο Κεφάλαιο 3, μια συνάρτηση σύνοψης λαμβάνει μια είσοδο δεδομένων οποιουδήποτε μεγέθους και παράγει μια σταθερού μεγέθους έξοδο, η οποία ονομάζεται σύνοψη (hash). Η σύνοψη αποτελεί μια σχεδόν μοναδική αναπαράσταση των δεδομένων εισόδου, που έχει σταθερό μήκος, ανεξάρτητα από το μέγεθος των δεδομένων που εισήχθησαν.

Ο ρόλος των συναρτήσεων σύνοψης στις αλυσίδες μπλοκ είναι πολυδιάστατος και καθοριστικός για την ασφάλεια και τη λειτουργία του δικτύου. Συγκεκριμένα, οι συναρτήσεις σύνοψης εξασφαλίζουν τα ακόλουθα:

- **Ακεραιότητα Δεδομένων:** Οι συναρτήσεις σύνοψης εξασφαλίζουν την ακεραιότητα των δεδομένων στην αλυσίδα μπλοκ. Στην κεφαλίδα κάθε μπλοκ υπάρχει η σύνοψη ολόκληρου του προηγούμενου μπλοκ. Εάν κάποιος προσπαθήσει να τροποποιήσει τα δεδομένα ενός μπλοκ, η σύνοψη του μπλοκ αυτού θα αλλάξει. Αυτό σημαίνει ότι το hash του τροποποιημένου μπλοκ δεν θα ταιριάζει πλέον με το hash που έχει καταχωρηθεί στο επόμενο μπλοκ, καθιστώντας έτσι την τροποποίηση προφανή και ανιχνεύσιμη. Με αυτό τον τρόπο, οι συναρτήσεις σύνοψης διασφαλίζουν ότι τα δεδομένα στην αλυσίδα μπλοκ παραμένουν αναλλοίωτα και αξιόπιστα.
- **Αμεταβλητότητα:** Οι συναρτήσεις σύνοψης παρέχουν την επικαλούμενη αμεταβλητότητα (immutability) μέσω της ιδιότητάς τους να παράγουν μοναδικές και μη αντιστρέψιμες εξόδους (one-way) για διαφορετικές εισόδους. Λαμβάνοντας υπόψη την ιδιότητας της αντίστασης πρώτου ορίσματος (pre-image resistance) των συναρτήσεων σύνοψης που τις καθιστά μονόδρομες (one-way), ακόμα κι αν κάποιος αποκτήσει τη σύνοψη των δεδομένων, δεν μπορεί να ανακατασκευάσει τα αρχικά δεδομένα. Επιπλέον, οι μικρές αλλαγές στην είσοδο οδηγούν σε δραματικές αλλαγές στη σύνοψη, γεγονός που προσθέτει ένα επίπεδο ασφάλειας.
- **Επικύρωση Συναλλαγών:** Κατά τη διαδικασία εξόρυξης (mining), οι εξορύκτες (miners) χρησιμοποιούν συναρτήσεις σύνοψης για να επιλύσουν ένα περίπλοκο μαθηματικό πρόβλημα, γνωστό ως απόδειξη εργασίας (proof of work). Αυτό το πρόβλημα περιλαμβάνει την εύρεση κατάλληλης τιμής εισόδου η οποία παράγει σύνοψη που είναι μικρότερη μιας οριακής τιμής, ή να πληροί άλλα κριτήρια, όπως η ύπαρξη ενός αριθμού μηδενικών στην αρχή της σύνοψης. Ο πρώτος εξορύκτης (κόμβος εξόρυξης) που θα βρει μια έγκυρη σύνοψη έχει το δικαίωμα να προσθέσει το νέο μπλοκ στην αλυσίδα και να λάβει την ανταμοιβή εξόρυξης.

Οι συναρτήσεις σύνοψης είναι ζωτικής σημασίας για την προστασία της αμεταβλητότητας του ψηφιακού καθολικού. Εάν η συνάρτηση σύνοψης που χρησιμοποιείται από μια αλυσίδα μπλοκ παραβιαστεί, ένας επιτιθέμενος θα μπορούσε να βρει διαφορετικά δεδομένα που παράγουν την ίδια τιμή σύνοψης, δηλαδή να εντοπίσει συγκρούσεις για κρίσιμες τιμές, όπως αυτές που αφορούν τα μπλοκ ή τα δέντρα Merkle. Αυτό θα επέτρεπε σε κακόβουλους κόμβους να ανακατασκευάσουν το ιστορικό της αλυσίδας μπλοκ και να προκαλέσουν την κατάρρευση του συστήματος.

Για αυτόν τον λόγο, η ασφάλεια των συναρτήσεων σύνοψης είναι ουσιαστική για την ασφάλεια της αλυσίδας μπλοκ. Αν δεν είναι επαρκής, μπορεί να απειληθεί η ασφάλεια της αλυσίδας μπλοκ αν λάβουμε υπόψη τις ακόλουθες περιπτώσεις:

- **Τρωτότητα της συνάρτησης:** Εάν βρεθεί κάποια αδυναμία σε μια συνάρτηση σύνοψης, μπορεί να γίνει εφικτή η εύρεση συγκρούσεων.
- **Μήκος σύνοψης:** Οι συναρτήσεις σύνοψης σχεδιάζονται ώστε ο καλύτερος τρόπος για να βρεθεί μια σύγκρουση να είναι μια εξαντλητική αναζήτηση (brute-force), με το χώρο αναζήτησης ίσο με το μέγεθος του χώρου των δυνατών εξόδων της συνάρτησης σύνοψης. Εάν ένας τέτοιος χώρος γίνει ουσιαστικά αναζητήσιμος – λόγω της χρήσης μιας συνάρτησης σύνοψης με πολύ μικρό μήκος εξόδου – τότε η συνάρτηση σύνοψης δεν είναι πλέον ανθεκτική στις συγκρούσεις και είναι ευάλωτη σε επιθέσεις.
- **Κβαντική υπολογιστική:** Ο αλγόριθμος Grover είναι ένας αλγόριθμος σχεδιασμένος για κβαντικούς υπολογιστές που μειώνει τον χώρο που πρέπει να αναζητήσει ένας επιτιθέμενος για να βρει μια σύγκρουση σύνοψης. Μόλις οι κβαντικοί υπολογιστές γίνονται πραγματικότητα, αυτό μπορεί να επιτρέψει την εύρεση συγκρούσεων για συναρτήσεις σύνοψης που εξακολουθούσαν να είναι ασφαλείς έναντι επιθέσεων από κλασικούς υπολογιστές.

11.3.2 Κρυπτογραφία Δημοσίου Κλειδιού

Η κρυπτογραφία δημοσίου κλειδιού αποτελεί ένα από τα θεμελιώδη στοιχεία της τεχνολογίας αλυσίδας μπλοκ καθώς συμβάλλει ουσιαστικά στην αυθεντικότητα των συναλλαγών, στην αυθεντικοποίηση και στην εξασφάλιση της ακεραιότητας των δεδομένων που υποβάλλονται για καταχώρηση στην αλυσίδα μπλοκ, και στην άρτια διευθυνσιοδότηση των συμμετεχόντων.

11.3.2.1 Ψηφιακές Υπογραφές

Οι ψηφιακές υπογραφές είναι ένα κρίσιμο στοιχείο για την ακεραιότητα και την αυθεντικοποίηση των συναλλαγών στην αλυσίδα μπλοκ. Μια ψηφιακή υπογραφή δημιουργείται με την χρήση του ιδιωτικού κλειδιού του υπογράφοντος και μπορεί να επαληθευτεί από οποιονδήποτε έχει το αντίστοιχο δημόσιο κλειδί. Αυτό επιτρέπει στους κόμβους του δικτύου να επιβεβαιώνουν ότι μια συναλλαγή προήλθε από τον νόμιμο κάτοχο του ιδιωτικού κλειδιού, διασφαλίζοντας έτσι την αυθεντικότητα των συναλλαγών.

Για παράδειγμα, όταν ένας χρήστης θέλει να πραγματοποιήσει μια συναλλαγή στο δίκτυο Bitcoin, υπογράφει τη συναλλαγή με το ιδιωτικό του κλειδί. Οι υπόλοιποι κόμβοι του δικτύου μπορούν να χρησιμοποιήσουν το δημόσιο κλειδί του χρήστη για να επαληθεύσουν την υπογραφή και να επιβεβαιώσουν ότι η συναλλαγή είναι γνήσια και δεν έχει τροποποιηθεί.

11.3.3 Διευθύνσεις

Οι διευθύνσεις στις αλυσίδες μπλοκ είναι μοναδικές ακολουθίες χαρακτήρων που χρησιμοποιούνται για την αναγνώριση και τη διαχείριση των περιουσιακών στοιχείων και των συναλλαγών στην αλυσίδα μπλοκ. Δημιουργούνται με τη χρήση κρυπτογραφικών αλγορίθμων (συνήθως συναρτήσεων σύνοψης) και συνδέονται με ένα ζεύγος κλειδιών: το δημόσιο και το ιδιωτικό κλειδί. Μια διεύθυνση μπορεί να αποτελεί τμήμα του δημόσιου κλειδιού του χρήστη ή της σύνοψης αυτού, και μπορεί να κοινοποιηθεί ελεύθερα, επιτρέποντας σε άλλους χρήστες να στέλνουν κρυπτονομίσματα ή άλλα ψηφιακά περιουσιακά στοιχεία στη συγκεκριμένη διεύθυνση. Το ιδιωτικό κλειδί, το οποίο πρέπει να διατηρείται μυστικό, είναι απαραίτητο για την υπογραφή των συναλλαγών και την πρόσβαση στα περιουσιακά στοιχεία που συνδέονται με τη διεύθυνση.

Κάθε διεύθυνση αντιπροσωπεύει έναν συγκεκριμένο χρήστη ή λογαριασμό και συνδέεται με τα κρυπτονομίσματα ή τα ψηφιακά περιουσιακά στοιχεία που ανήκουν σε αυτόν. Μέσω των διευθύνσεων, οι χρήστες μπορούν να διαχειρίζονται τα κεφάλαια τους και να παρακολουθούν τις συναλλαγές τους στο δίκτυο. Επίσης, οι διευθύνσεις παρέχουν ασφάλεια στις συναλλαγές, καθώς η υπογραφή μιας συναλλαγής με το ιδιωτικό κλειδί αποδεικνύει ότι ο αποστολέας είναι ο νόμιμος κάτοχος των περιουσιακών στοιχείων που αναφέρονται σε μια συναλλαγή. Οι υπόλοιποι κόμβοι του δικτύου χρησιμοποιούν το δημόσιο κλειδί για να επαληθεύσουν την υπογραφή και την εγκυρότητα της συναλλαγής.

Επιπλέον, οι διευθύνσεις προσφέρουν ένα επίπεδο απορρήτου και ανωνυμίας στους χρήστες. Παρόλο που οι συναλλαγές στην αλυσίδα μπλοκ είναι δημόσιες, οι διευθύνσεις δεν περιέχουν προσωπικές πληροφορίες, επιτρέποντας στους χρήστες να παραμένουν ανώνυμοι. Ωστόσο, με την ανάλυση των συναλλαγών, μπορεί να είναι δυνατή η σύνδεση μιας διεύθυνσης με έναν συγκεκριμένο χρήστη. Οι χρήστες μπορούν να δημιουργούν νέες διευθύνσεις για διαφορετικούς σκοπούς, ακόμα και για κάθε νέα συναλλαγή, διευκολύνοντας έτσι τη διαχείριση των συναλλαγών τους. Το σύνολο των διευθύνσεων του χρήστη μπορεί να αποθηκεύεται στο ψηφιακό πορτοφόλι του.

Ένα παράδειγμα διεύθυνσης αλυσίδας μπλοκ είναι οι διευθύνσεις Bitcoin, οι οποίες παράγονται με τη χρήση συναρτήσεων σύνοψης: η διεύθυνση αποτελεί τη σύνοψη με RIPEMD, μήκους 160 bits, της σύνοψης με SHA-256 του δημοσίου κλειδιού του χρήστη. Μια διεύθυνση Bitcoin μπορεί να μοιάζει με αυτή: **1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa**. Στο Ethereum, η διεύθυνση ενός πορτοφολιού είναι τα τελευταία 20 bytes της σύνοψης του δημοσίου κλειδιού του χρήστη. Αυτό επιτρέπει στους χρήστες να λαμβάνουν νομίσματα Ether με ασφάλεια, γνωρίζοντας ότι μόνο αυτοί μπορούν να έχουν πρόσβαση στα κεφάλαια τους.

11.3.4 Έξυπνα Συμβόλαια

Ένα από τα πιο σημαντικά χαρακτηριστικά που υποστηρίζονται από πολλές αλυσίδες μπλοκ είναι τα έξυπνα συμβόλαια (smart contract). Τα έξυπνα συμβόλαια είναι τμήματα λογισμικού που εκτελούν συγκεκριμένες ενέργειες βάσει της κατάστασης του συστήματος ή μιας συναλλαγής που λαμβάνει χώρα [2]. Αυτά τα τμήματα κώδικα εφαρμόζουν, επαληθεύουν ή εκτελούν τους όρους μιας συμφωνίας αυτόματα, χωρίς την ανάγκη ανθρώπινης παρέμβασης. Παραδείγματα τέτοιων έξυπνων συμβολαίων περιλαμβάνουν τα smart contracts του Ethereum και τα chaincodes του Hyperledger Fabric.

Η εκτέλεση ενός έξυπνου συμβολαίου γίνεται από τους κόμβους του δικτύου της αλυσίδας μπλοκ. Όλοι οι κόμβοι που συμμετέχουν στην εκτέλεσή του πρέπει να καταλήξουν στα ίδια αποτελέσματα, τα οποία στη συνέχεια καταγράφονται στην αλυσίδα μπλοκ.

Δεδομένου ότι ο κώδικας των έξυπνων συμβολαίων αποθηκεύεται στην αλυσίδα μπλοκ, οποιαδήποτε απόπειρα παραβίασης γίνεται άμεσα ανιχνεύσιμη, καθιστώντας τον κώδικα ανθεκτικό σε παραβιάσεις. Αυτό το χαρακτηριστικό επιτρέπει στα έξυπνα συμβόλαια να λειτουργούν ως αξιόπιστα τρίτα μέρη, επαληθεύοντας και εκτελώντας τους όρους των συμβολαίων χωρίς την ανάγκη μεσολάβησης ανθρώπων ή οργανισμών.

11.4 Αλγόριθμος Συναίνεσης

Στις αλυσίδες μπλοκ, η εμπιστοσύνη μεταξύ των συμμετεχόντων βασίζεται σε κανόνες ή αλγορίθμους συναίνεσης (consensus algorithm) που ακολουθούν όλοι οι συμμετέχοντες για την επαλήθευση, επικύρωση και προσθήκη συναλλαγών στην αλυσίδα μπλοκ. Οι αλγόριθμοι συναίνεσης διασφαλίζουν ότι τα καθολικά ενημερώνονται με τις ίδιες συναλλαγές και με την ίδια σειρά συναλλαγών για όλους τους συμμετέχοντες και μόνο όταν οι συναλλαγές εγκρίνονται από τους κατάλληλους συμμετέχοντες. Με αυτόν τον τρόπο, επιτρέπουν σε ομάδες χρηστών όπου δεν υπάρχει εμπιστοσύνη, να συνεργαστούν.

Οι πιο συχνά χρησιμοποιούμενοι αλγόριθμοι συναίνεσης σε αλυσίδες μπλοκ παρουσιάζονται στις ακόλουθες υποενότητες.

11.4.1 Απόδειξη Εργασίας

Η Απόδειξη Εργασίας (Proof of Work – PoW) είναι ο πρώτος αλγόριθμος συναίνεσης που εφαρμόστηκε επιτυχώς σε αλυσίδες μπλοκ και αποτελεί τον ακρογωνιαίο λίθο της ασφάλειας του Bitcoin. Η βασική ιδέα πίσω από το PoW είναι ότι οι κόμβοι του δικτύου, γνωστοί ως εξορύκτες (miners), ανταγωνίζονται για να λύσουν ένα δύσκολο μαθηματικό πρόβλημα. Ο πρώτος που βρίσκει τη λύση δημιουργεί ένα νέο μπλοκ και το προσθέτει στην αλυσίδα μπλοκ, κερδίζοντας μια ανταμοιβή σε κρυπτονόμισμα. Η λύση πρέπει να είναι δύσκολη να βρεθεί, και έτσι απαιτεί αρκετή υπολογιστική ισχύ, αλλά εύκολη να επαληθευτεί από το δίκτυο.

Ο αλγόριθμος λειτουργεί ως εξής: ο εξορύκτης (κόμβος) συλλέγει μια ομάδα από τις νέες συναλλαγές που έχουν σταλεί στο δίκτυο και τις ομαδοποιεί σε ένα μπλοκ. Στη συνέχεια, ο εξορύκτης πρέπει να βρει μια τιμή σύνοψης (hash value) για αυτό το μπλοκ που να πληροί συγκεκριμένα κριτήρια, τα οποία καθορίζονται από το πρωτόκολλο του δικτύου. Η διαδικασία εύρεσης αυτής της τιμής απαιτεί δοκιμές πολλών διαφορετικών εισόδων, γνωστών ως *nonce*, μέχρι να βρεθεί η σωστή τιμή.

Το μαθηματικό πρόβλημα που πρέπει να λύσουν οι εξορύκτες βασίζεται στη συνάρτηση σύνοψης που εφαρμόζεται στην κεφαλίδα του μπλοκ. Οι εξορύκτες προσπαθούν να βρουν μια τυχαία τιμή *nonce* έτσι ώστε η σύνοψη της κεφαλίδας του μπλοκ μαζί με την τιμή *nonce* να είναι μικρότερη ή ίση από μια προκαθορισμένη τιμή στόχο (target), δηλαδή:

$$H(\text{Block_Header} + \text{nonce}) \leq \text{Target}$$

όπου H είναι η συνάρτηση σύνοψης και Block_Header είναι τα δεδομένα της κεφαλίδας του μπλοκ. Το Target είναι η τιμή στόχος, η οποία καθορίζεται από τη δυσκολία του δικτύου. Όσο πιο μικρή είναι η τιμή στόχος, τόσο πιο δύσκολο είναι να βρεθεί μια αποδεκτή σύνοψη.

Ο πρώτος εξορύκτης που θα βρει τη σωστή τιμή σύνοψης έχει το δικαίωμα να προσθέσει το δικό του νέο μπλοκ στην αλυσίδα και να λάβει μια ανταμοιβή σε κρυπτονομίσματα, καθώς και τα τέλη των συναλλαγών που περιλαμβάνονται στο μπλοκ. Αυτή η διαδικασία διασφαλίζει τη συναίνεση στο δίκτυο και ταυτόχρονα αποτελεί μηχανισμό διανομής νέων κρυπτονομισμάτων στους εξορύκτες.

Η ασφάλεια του δικτύου εξασφαλίζεται από το γεγονός ότι η εύρεση της σωστής τιμής είναι χρονοβόρα και υπολογιστικά απαιτητική, ενώ η επαλήθευση του αποτελέσματος είναι γρήγορη και εύκολη για το υπόλοιπο δίκτυο. Ο μόνος τρόπος για την εύρεση της σωστής *nonce* είναι μέσω της δοκιμής πολλών πιθανών τιμών μέχρι να παραχθεί μια σύνοψη που πληροί το κριτήριο της τιμής στόχου.

Το κύριο πλεονέκτημα της Απόδειξης Εργασίας είναι η ασφάλεια που παρέχει στο δίκτυο. Η ανάγκη για σημαντική υπολογιστική ισχύ καθιστά τις επιθέσεις πολύ δαπανηρές και δύσκολες να επιτευχθούν. Ειδικότερα, μια επίθεση του 51%, όπου ένας κακόβουλος παράγοντας θα αποκτούσε τον έλεγχο της πλειοψηφίας της υπολογιστικής ισχύος του δικτύου, είναι εξαιρετικά δύσκολη και κοστοβόρα. Αυτό αποτρέπει τους κακόβουλους παράγοντες από το να επιχειρούν να παραποίησουν την αλυσίδα.

Ωστόσο, η Απόδειξη Εργασίας έχει δεχθεί κριτική για την υψηλή ενεργειακή κατανάλωση που απαιτεί. Η συνεχής ανάγκη για υπολογιστική ισχύ και η ενέργεια που κατανάλωνται για την επίλυση των προαναφερθέντων μαθηματικών προβλημάτων έχουν σημαντικό περιβαλλοντικό αντίκτυπο. Για παράδειγμα, οι εξορύκτες του Bitcoin χρησιμοποιούν κατά προσέγγιση 1100MW ανά δευτερόλεπτο κατά τη διαδικασία της εξόρυξης [3]. Επιπλέον, η διαδικασία εξόρυξης μπορεί να οδηγήσει σε συγκέντρωση της υπολογιστικής ισχύος σε λίγες μεγάλες εξορυκτικές κοινοπραξίες (mining pools), γεγονός που μπορεί να μειώσει την αποκέντρωση και την ανθεκτικότητα του δικτύου.

Παρά τα μειονεκτήματα αυτά, η Απόδειξη Εργασίας παραμένει ένας από τους πιο αξιόπιστους και ασφαλείς αλγόριθμους συναίνεσης, χρησιμοποιούμενος ευρέως σε πολλά κρυπτονομίσματα πέραν του Bitcoin, όπως το Litecoin και το Bitcoin Cash. Η διαρκής εξέλιξη των τεχνολογιών εξόρυξης και η αναζήτηση λύσεων για τη μείωση της ενεργειακής κατανάλωσης συνεχίζουν να βελτιώνουν τον αλγόριθμο PoW, διατηρώντας τον ως βασικό εργαλείο για την επίτευξη συναίνεσης στις αλυσίδες μπλοκ.

11.4.2 Απόδειξη Συμμετοχής

Η Απόδειξη Συμμετοχής (Proof of Stake – PoS) είναι ένας εναλλακτικός αλγόριθμος συναίνεσης που προτάθηκε ως λύση για την υψηλή ενεργειακή κατανάλωση Της Απόδειξης Εργασίας [roof_of_stake]. Η βασική ιδέα πίσω από την Απόδειξη Συμμετοχής είναι ότι η πιθανότητα ενός κόμβου να επιλεγεί για τη δημιουργία του επόμενου μπλοκ εξαρτάται από την ποσότητα των κρυπτονομισμάτων που κατέχει και έχει δεσμεύσει ως εγγύηση στο δίκτυο.

Στον αλγόριθμο PoS, οι κόμβοι που επιθυμούν να συμμετάσχουν στη διαδικασία συναίνεσης πρέπει να «κλειδώσουν» ένα συγκεκριμένο ποσό κρυπτονομισμάτων ως εγγύηση. Αυτό το ποσό λειτουργεί ως «μερίδιο» (stake). Οι κόμβοι με μεγαλύτερο μερίδιο έχουν περισσότερες πιθανότητες να επιλεγούν για την προσθήκη

του επόμενου μπλοκ. Αυτό το σύστημα ενθαρρύνει τη διατήρηση της ασφάλειας του δικτύου, καθώς οι κόμβοι με μεγάλο μερίδιο έχουν κίνητρο να διατηρούν την αλυσίδα μπλοκ ασφαλή και σταθερή.

Η διαδικασία επιλογής των κόμβων μπορεί να γίνει με διάφορους τρόπους, όπως τυχαία επιλογή με βάση το μερίδιο ή συνδυασμός ηλικίας και ποσότητας του μεριδίου. Ανεξάρτητα από τον ακριβή μηχανισμό, ο βασικός στόχος είναι να εξασφαλιστεί ότι οι κόμβοι με μεγαλύτερο μερίδιο έχουν περισσότερες πιθανότητες να δημιουργήσουν νέα μπλοκ, αλλά να παραμένει ένα στοιχείο τυχαιότητας για την αποφυγή πλήρους συγκέντρωσης της εξουσίας.

Ένα από τα κύρια πλεονεκτήματα του PoS είναι η σημαντική μείωση της ενεργειακής κατανάλωσης σε σύγκριση με το PoW. Επειδή οι κόμβοι δεν χρειάζεται να λύσουν πολύπλοκα μαθηματικά παζλ, η διαδικασία συναίνεσης απαιτεί πολύ λιγότερη υπολογιστική ισχύ και ενέργεια. Αυτό καθιστά το PoS μια πιο φιλική προς το περιβάλλον λύση.

Ωστόσο, το PoS έχει και ορισμένα μειονεκτήματα. Ένα από τα κύρια ζητήματα είναι η πιθανότητα συγκέντρωσης της εξουσίας σε κόμβους που κατέχουν μεγάλα ποσά κρυπτονομίσματων. Αυτή η συγκέντρωση μπορεί να οδηγήσει σε «κεντρικοποίηση» του δικτύου, κάτι που αντίκειται στην αρχή της αποκέντρωσης που διέπει τις αλυσίδες μπλοκ. Επιπλέον, υπάρχουν ανησυχίες για την ασφάλεια του PoS, όπως η πιθανότητα επιθέσεων από κακόβουλους κόμβους που προσπαθούν να επηρεάσουν τη διαδικασία συναίνεσης.

Το Ethereum, ένα από τα πιο γνωστά κρυπτονομίσματα, έχει υιοθετήσει το PoS ως αλγόριθμο συναίνεσης, υποστηρίζοντας τη μετάβαση από το PoW σε μια πιο ενεργειακά αποδοτική λύση. Άλλα κρυπτονομίσματα, όπως το Cardano και το Tezos, χρησιμοποιούν επίσης το PoS, αναδεικνύοντας τη δημοτικότητα και την αποδοτικότητα αυτού του αλγορίθμου συναίνεσης.

11.4.3 (Πρακτική) Βυζαντινή Ανοχή Σφαλμάτων

Η έννοια της Βυζαντινής Ανοχής Σφαλμάτων (Byzantine Fault Tolerance – BFT) αναφέρεται στην ικανότητα ενός κατανεμημένου συστήματος να συνεχίσει να λειτουργεί σωστά και να επιτυγχάνει συμφωνία μεταξύ των συμμετεχόντων κόμβων, ακόμη και όταν μερικοί από αυτούς αποτυγχάνουν ή συμπεριφέρονται κακόβουλα. Ο όρος «Βυζαντινή» προέρχεται από το πρόβλημα των Βυζαντινών Στρατηγών, ένα κλασικό πρόβλημα στην πληροφορική που περιγράφει την δυσκολία επίτευξης συμφωνίας σε ένα κατανεμημένο σύστημα όπου οι κόμβοι μπορούν να επικοινωνούν μέσω ενός αναξιόπιστου δικτύου.

Συγκεκριμένα, το πρόβλημα των Βυζαντινών Στρατηγών αφορά μια ομάδα στρατηγών που πρέπει να συντονιστούν για να επιτεθούν ή να υποχωρήσουν. Κάποιοι από αυτούς μπορεί να είναι προδότες και να στέλνουν παραπλανητικά μηνύματα. Το σύστημα πρέπει να διασφαλίσει ότι οι πιστοί στρατηγοί μπορούν να συμφωνήσουν σε ένα κοινό σχέδιο, ανεξάρτητα από τις κακόβουλες ενέργειες των προδοτών.

Η Βυζαντινή Ανοχή Σφαλμάτων απαιτεί την ύπαρξη ενός αλγορίθμου που επιτρέπει στο σύστημα να επιτύχει ομόφωνη συμφωνία, εφόσον λιγότερο από το ένα τρίτο των κόμβων είναι κακόβουλοι.

Η Πρακτική Βυζαντινή Ανοχή Σφαλμάτων (Practical Byzantine Fault Tolerance – PBFT) είναι μια πρακτική υλοποίηση της θεωρητικής έννοιας της BFT που αναπτύχθηκε για να είναι εφαρμόσιμη σε πραγματικά κατανεμημένα συστήματα, όπως αυτά που χρησιμοποιούνται στις αλυσίδες μπλοκ (blockchain). Ο αλγόριθμος PBFT προτάθηκε από τον Miguel Castro και τον Barbara Liskov το 1999 [4] και σχεδιάστηκε για να παρέχει ανοχή σε Βυζαντινά σφάλματα με αποδοτικό τρόπο.

Ο αλγόριθμος PBFT λειτουργεί με βάση την έννοια της «προετοιμασίας» και της «δέσμευσης» των μηνυμάτων, χρησιμοποιώντας μια σειρά από φάσεις για να εξασφαλίσει ότι όλοι οι κόμβοι συμφωνούν στην ίδια κατάσταση του συστήματος. Η διαδικασία του PBFT περιλαμβάνει τα ακόλουθα βήματα:

- **Προετοιμασία (Prepare):** Κάθε κόμβος λαμβάνει και επεξεργάζεται την αίτηση (request) από τον πελάτη. Στη συνέχεια, αποστέλλει ένα μήνυμα προετοιμασίας σε όλους τους άλλους κόμβους.
- **Προ-δέσμευση (Pre-prepare):** Αφού λάβει τα μηνύματα προετοιμασίας από τους άλλους κόμβους, ένας κόμβος στέλνει ένα μήνυμα προ-δέσμευσης σε όλους τους υπόλοιπους.

- Δέσμευση (Commit): Όταν ένας κόμβος λάβει αρκετά μηνύματα προ-δέσμευσης, στέλνει ένα μήνυμα δέσμευσης σε όλους τους άλλους κόμβους.
- Αποδοχή (Reply): Όταν ένας κόμβος λάβει αρκετά μηνύματα δέσμευσης, θεωρεί ότι η κατάσταση του συστήματος είναι επιβεβαιωμένη και αποστέλλει μια απάντηση στον πελάτη.

Το βασικό πλεονέκτημα του αλγορίθμου PBFT είναι η υψηλή αξιοπιστία καθώς μπορεί να λειτουργήσει σωστά αν λιγότερο από το ένα τρίτο των κόμβων είναι κακόβουλοι. Επιπλέον, σε σύγκριση με άλλους αλγόριθμους συναίνεσης, ο αλγόριθμος PBFT είναι πιο αποδοτικός σε όρους χρόνου και πόρων, καθιστώντας τον ιδανικό για συστήματα με υψηλές απαιτήσεις απόδοσης.

Ο αλγόριθμος PBFT ωστόσο έχει και κάποια μειονεκτήματα. Η πολυπλοκότητα της επικοινωνίας αυξάνεται δραματικά καθώς ο αριθμός των κόμβων αυξάνεται, γεγονός που μπορεί να επηρεάσει την απόδοση σε μεγάλα δίκτυα. Επιπλέον, ο PBFT απαιτεί σημαντικούς υπολογιστικούς και δικτυακούς πόρους για να λειτουργήσει αποτελεσματικά, ιδιαίτερα σε δίκτυα με υψηλή αλληλεπίδραση.

11.4.4 Απόδειξη Αρχής

Η Απόδειξη Αρχής (Proof of Authority – PoA) αποτελεί ακόμη έναν από σημαντικό αλγόριθμο συναίνεσης το κύριο χαρακτηριστικό του οποίου είναι πως βασίζεται στην ταυτότητα και την αξιοπιστία των επικυρωτών (validators). Στην PoA, η εξουσιοδότηση των επικυρωτών εξασφαλίζει την ασφάλεια και την ακεραιότητα του δικτύου.

Η βασική αρχή της PoA είναι ότι οι κόμβοι που επικυρώνουν τις συναλλαγές και δημιουργούν νέα μπλοκ στο δίκτυο είναι προκαθορισμένοι και ταυτοποιημένοι. Οι επικυρωτές αυτοί επιλέγονται με βάση την αξιοπιστία και την εμπιστοσύνη που έχουν αποκτήσει στην κοινότητα. Η ταυτότητά τους είναι γνωστή και επαληθεύσιμη, γεγονός που δημιουργεί ένα ισχυρό κίνητρο για αυτούς να συμπεριφέρονται με ειλικρίνεια και να διατηρούν την ακεραιότητα του δικτύου.

Σε κάθε κύκλο δημιουργίας μπλοκ, ένας από τους επικυρωτές επιλέγεται για να δημιουργήσει το νέο μπλοκ. Η επιλογή αυτή μπορεί να γίνεται με διάφορους τρόπους, όπως σε σειριακή ή τυχαία βάση. Η PoA είναι ιδιαίτερα αποδοτική και ταχεία σε σύγκριση με άλλους αλγορίθμους συναίνεσης, διότι δεν απαιτεί έντονη υπολογιστική εργασία ή μεγάλο αριθμό επικυρωτών.

Η PoA παρουσιάζει αρκετά πλεονεκτήματα. Πρώτον, επιτρέπει τη γρήγορη επεξεργασία των συναλλαγών και την άμεση δημιουργία μπλοκ, γεγονός που αυξάνει την απόδοση του δικτύου. Δεύτερον, σε αντίθεση με το PoW, η PoA δεν απαιτεί σημαντική κατανάλωση ενέργειας, καθιστώντας την πιο περιβαλλοντικά φιλική. Επιπλέον, η διαχείριση του δικτύου είναι ευκολότερη, καθώς οι επικυρωτές είναι προκαθορισμένοι και μπορούν να ελέγχονται άμεσα από την κοινότητα.

Ωστόσο, υπάρχουν και μειονεκτήματα. Η PoA μπορεί να οδηγήσει σε «συγκεντροποίηση», καθώς οι εξουσιοδοτημένοι επικυρωτές έχουν μεγάλη ισχύ και έλεγχο στο δίκτυο. Η ασφάλεια του δικτύου εξαρτάται από την αξιοπιστία των επικυρωτών. Εάν οι επικυρωτές συμπεριφέρονται κακόβουλα, το δίκτυο μπορεί να υποστεί ζημιά. Επιπλέον, η επιλογή και η συντήρηση αξιόπιστων επικυρωτών μπορεί να καταστεί δύσκολη καθώς το δίκτυο μεγαλώνει.

11.5 Δημοφιλείς Πλατφόρμες Αλυσίδων Μπλοκ

Από τις πιο δημοφιλείς και αντιπροσωπευτικές πλατφόρμες αλυσίδων μπλοκ είναι αυτές του Bitcoin, Ethereum και Hyperledger Fabric, τα βασικά χαρακτηριστικά των οποίων αναλύονται σε αυτήν την ενότητα.

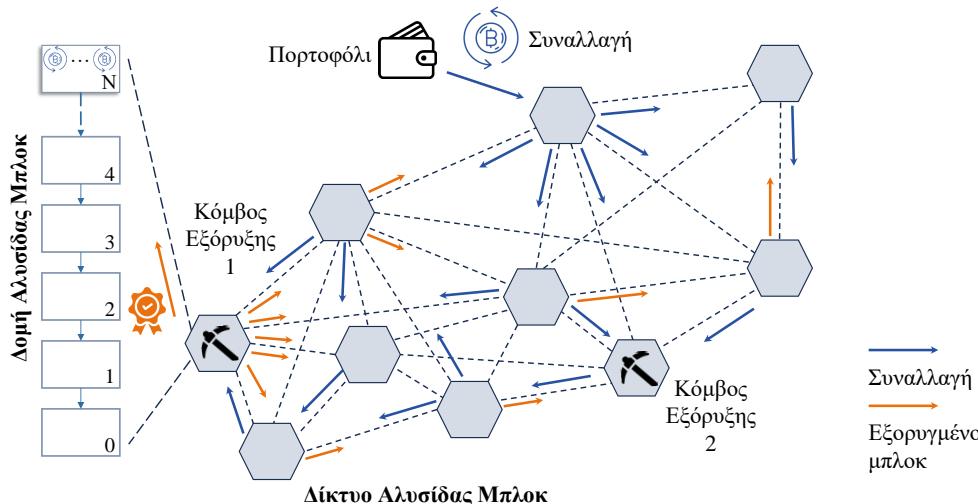
11.5.1 Bitcoin

Το Bitcoin είναι το πρώτο ψηφιακό νόμισμα που αξιοποιεί την τεχνολογία δημόσιων αλυσίδων μπλοκ χωρίς άδεια και δημιουργήθηκε το 2009 από τον Satoshi Nakamoto [5], ψευδώνυμο ενός απόμουν ή ομάδας απόμων

που το σχεδίασαν και το υλοποίησαν. Στο Bitcoin όλα τα νομίσματα (BTC) δημιουργούνται, ξοδεύονται και μεταφέρονται ψηφιακά μέσα στο οικοσύστημα του Bitcoin. Αυτό που το κάνει ζεχωριστό είναι ότι δεν υπάρχει κάποια κεντρική οντότητα που να δημιουργεί νομίσματα και να επαληθεύει συναλλαγές. Αντιθέτως, οι κόμβοι που ανήκουν στο δίκτυο Bitcoin αναλαμβάνουν αυτόν τον ρόλο.

Στο σύστημα του Bitcoin, οι χρήστες εκπροσωπούνται από διευθύνσεις, οι οποίες δημιουργούνται βάσει των δημοσίων κλειδιών των χρηστών και διατηρούνται σε ψηφιακά πορτοφόλια, μαζί με τα αντίστοιχα ιδιωτικά κλειδιά του κατόχου τους. Η απευθείας ταυτοποίηση του κατόχου δεν είναι εφικτή και για αυτό οι διευθύνσεις αυτές αποτελούν ένα είδος ψευδωνύμου.

Για την πραγματοποίηση μιας συναλλαγής (Σχήμα 11.5), έστω μεταξύ δύο χρηστών A και B όπου ο A θέλει να στείλει 2 BTCs στον B, ο A υπογράφει ψηφιακά με χρήση του ιδιωτικού κλειδιού του τη συναλλαγή και τη μεταδίδει σε ολόκληρο το δίκτυο της αλυσίδας μπλοκ του Bitcoin. Αυτή η συναλλαγή παραμένει εκκρεμής έως ότου ένας εξορύκτης την συμπεριλάβει σε ένα νέα μπλοκ της αλυσίδας. Οι κόμβοι που δικτύουν συλλέγουν τις όποιες εκκρεμείς συναλλαγές και στη συνέχεια επιβεβαιώνουν την ορθότητά τους πριν τις επικυρώσουν. Σύμφωνα με τον αλγόριθμο συναίνεσης PoW, ο εξορύκτης που θα βρει τη λύση στο μαθηματικό πρόβλημα, τη διαμοιράζει μέσα από ένα νέο μπλοκ στο υπόλοιπο δίκτυο, το οποίο επαληθεύει την ορθότητά της και εάν το μπλοκ είναι έγκυρο προστίθεται στο τέλος της αλυσίδας μπλοκ. Με αυτόν τον τρόπο οι συναλλαγές στο μπλοκ επικυρώνονται, μεταξύ αυτών ο A αποστέλλει στον B 2 BTCs και ο εξορύκτης ανταμείβεται με τα τέλη συναλλαγής (transaction fee) που αφορούν στις συναλλαγές που επικυρώθηκαν εντός του νέου μπλοκ και με μερικά νέα BTC από τα 21 εκατομμύρια BTCs που προβλέπεται να εξορυχθούν συνολικά. Αυτή η διαδικασία της ανταμοιβής προσαρμόζεται με την πάροδο του χρόνου. Όταν κυκλοφόρησε για πρώτη φορά το Bitcoin, το μέρος της ανταμοιβής με νέα BTC ήταν 50 BTCs. Έκτοτε, κάθε τέσσερα χρόνια μειώνεται στο μισό. Η διαδικασία αυτή ονομάζεται “halving”. Μετά το πρόσφατο halving το 2024, η ανταμοιβή εξόρυξης για το Bitcoin είναι 3.125 BTC ανά μπλοκ. Οι επιπτώσεις του halving είναι πολυδιάστατες, επηρεάζοντας κυρίως την κερδοφορία των εξορυκτών. Οι εξορύκτες με υψηλότερο κόστος ενέργειας ή λιγότερο αποδοτικό εξοπλισμό ενδέχεται να βρεθούν σε δυσκολία να παραμείνουν κερδοφόροι, ειδικά αν η τιμή του Bitcoin δεν αυξηθεί σημαντικά για να αντισταθμίσει τη μειωμένη ανταμοιβή



Σχήμα 11.5: Επισκόπηση του συστήματος Bitcoin.

Ο έλεγχος εγκυρότητας μιας συναλλαγής διασφαλίζει δύο κύρια ζητήματα. Το πρώτο είναι ότι ο A έχει στην κατοχή του τον αριθμό των BTCs που απαιτούνται από την συναλλαγή. Αυτό επαληθεύεται εξετάζοντας το ιστορικό του στην συνολική αλυσίδα μπλοκ και διασταυρώνοντας τον τρόπο με τον οποίο ο A έλαβε αυτά τα BTCs. Το δεύτερο είναι ότι ο A δεν μπορεί να πραγματοποιήσει την ίδια συναλλαγή δύο φορές στέλνοντας τα ίδια κρυπτονομίσματα σε δύο διαφορετικούς παραλήπτες ταυτόχρονα, γνωστό και ως επίθεση διπλής δαπάνης (Ενότητα 11.6.6).

Σύμφωνα με το πρωτόκολλο του Bitcoin, ένα μπλοκ εξορύσσεται κατά μέσο όρο, κάθε 10 λεπτά. Για να διατηρηθεί σταθερός αυτός ο ρυθμός, ο βαθμός δυσκολίας του μαθηματικού προβλήματος αναπροσαρμόζεται ανά τακτά χρονικά διαστήματα. Μετά την επιτυχή εξόρυξη 2.016 μπλοκ (περίπου κάθε δύο εβδομάδες), το σύστημα υπολογίζει το μέσο χρόνο εξόρυξης τους. Εάν είναι λιγότερο από 10 λεπτά, ο βαθμός δυσκολίας αυξάνεται. Εάν είναι περισσότερο από 10 λεπτά, ο βαθμός δυσκολίας μειώνεται.

11.5.2 Ethereum

Το Ethereum [6] ανήκει και αυτό στην κατηγορία αλυσίδων μπλοκ χωρίς άδεια. Είναι μια καινοτόμος πλατφόρμα αλυσίδας μπλοκ που εισήγαγε τον κόσμο στις απεριόριστες δυνατότητες της αποκεντρωμένης τεχνολογίας. Δημιουργήθηκε το 2015 με στόχο να επεκτείνει τη λειτουργικότητα που είχε εισάγει το Bitcoin, παρέχοντας ένα πιο ευέλικτο εργαλείο για τη δημιουργία προγραμματιζόμενων εφαρμογών. Ενώ το Bitcoin εστιάζει κυρίως στις συναλλαγές χρημάτων, το Ethereum στοχεύει στη δημιουργία μιας ανοικτής πλατφόρμας που επιτρέπει στους προγραμματιστές να γράφουν έξυπνα συμβόλαια και να δημιουργούν αποκεντρωμένες εφαρμογές (Decentralized Applications – dApps).

Έτσι, το Ethereum μπορεί να χαρακτηριστεί ως ένα πλήρες οικοσύστημα που υποστηρίζει τη δημιουργία και εκτέλεση έξυπνων συμβολαίων, τα οποία είναι εκτελούνται αυτόματα όταν πληρούνται οι προκαθορισμένοι όροι, χωρίς να απαιτείται ανθρώπινη παρέμβαση ή η εμπλοκή μεσαζόντων, όπως δικηγόροι ή τράπεζες.

Η πλατφόρμα τροφοδοτείται από το κρυπτονόμισμα Ether (ETH), το οποίο λειτουργεί ως «καύσιμο» για τις συναλλαγές και την εκτέλεση έξυπνων συμβολαίων. Οι χρήστες πληρώνουν τέλη, γνωστά ως gas fees για την εκτέλεση των συναλλαγών και των συμβολαίων, και αυτά τα τέλη καθορίζονται από την πολυπλοκότητα της πράξης και τη συμφόρηση του δικτύου.

Η διαδικασία μιας συναλλαγής στο Ethereum είναι μια περίπλοκη διαδικασία, καθώς μπορεί να περιλαμβάνει την αλληλεπίδραση με έξυπνα συμβόλαια ή αποκεντρωμένες εφαρμογές. Ο κύκλος ζωής μιας συναλλαγής αποτελείται από τα εξής στάδια:

- **Δημιουργία συναλλαγής:** Ο χρήστης δημιουργεί μια συναλλαγή που περιλαμβάνει είτε την αποστολή Ether σε μια άλλη διεύθυνση, είτε την εκτέλεση ενός έξυπνου συμβολαίου. Αυτή η συναλλαγή συνοδεύεται από ένα ποσό “gas”, που λειτουργεί ως αμοιβή για την εκτέλεση.
- **Υποβολή στο δίκτυο:** Η συναλλαγή αποστέλλεται στο Ethereum δίκτυο και περιμένει να επικυρωθεί από τους επικυρωτές (validators), οι οποίοι εξετάζουν την εγκυρότητα της συναλλαγής και, αν όλα είναι εντάξει, τη συμπεριλαμβάνουν σε ένα μπλοκ.
- **Εκτέλεση έξυπνου συμβολαίου:** Εάν η συναλλαγή περιλαμβάνει έξυπνο συμβόλαιο, τότε το Ethereum Virtual Machine (EVM) αναλαμβάνει την εκτέλεσή του. Το EVM είναι το σύστημα που επεξεργάζεται τα συμβόλαια και τις εντολές στο Ethereum, διασφαλίζοντας ότι τα πάντα γίνονται σωστά και με ασφάλεια.
- **Συμπερίληψη σε μπλοκ:** Η συναλλαγή εισάγεται σε ένα νέο μπλοκ της αλυσίδας. Το Ethereum δημιουργεί και επικυρώνει νέα blocks κάθε λίγα δευτερόλεπτα, καθιστώντας το δίκτυο αρκετά γρήγορο.
- **Τελική επικύρωση:** Αναφέρεται στο σημείο όπου μια συναλλαγή θεωρείται οριστικά επικυρωμένη και δεν μπορεί να αναστραφεί ή να αλλάξει. Αυτό επιτυγχάνεται όταν μια συναλλαγή περιλαμβάνεται σε αρκετά μπλοκ μετά το αρχικό μπλοκ στο οποίο ενσωματώθηκε, συνήθως μετά από 12 επιβεβαιώσεις. Το πρωτόκολλο GHOST διασφαλίζει ότι η «βαρύτερη» αλυσίδα επικρατεί και γίνεται η κύρια αλυσίδα του δικτύου, εξασφαλίζοντας την αμετάκλητη καταγραφή των συναλλαγών. Με την έλευση του Ethereum 2.0 η διαδικασία της τελικής επικύρωσης γίνεται πιο γρήγορη και αποδοτική, ενισχύοντας την ασφάλεια του δικτύου.

Το Ethereum έχει αναπτυχθεί με τρόπο ώστε να διασφαλίζει τη γρήγορη δημιουργία μπλοκ. Ο χρόνος μεταξύ της δημιουργίας μπλοκ κυμαίνεται κατά μέσο όρο μεταξύ 13 και 15 δευτερολέπτων, γεγονός που το

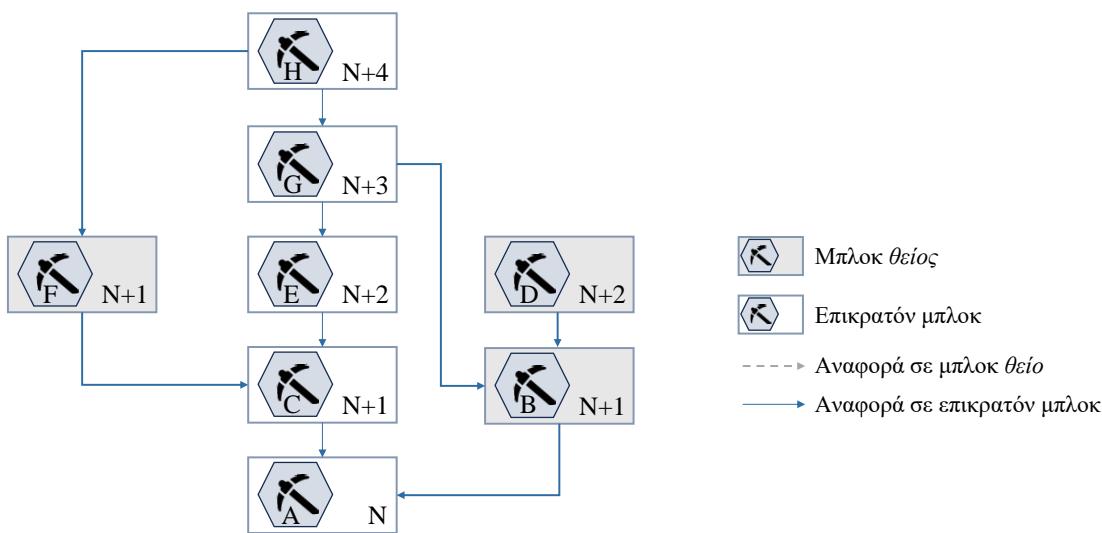
καθιστά πολύ ταχύτερο σε σύγκριση με το Bitcoin, όπου χρειάζονται περίπου 10 λεπτά για τη δημιουργία ενός νέου μπλοκ. Ωστόσο, η γρήγορη παραγωγή μπλοκ μπορεί να οδηγήσει στην ταυτόχρονη δημιουργία ανταγωνιστικών μπλοκ, τα οποία εξορύσσονται σχεδόν ταυτόχρονα από διαφορετικούς κόμβους του δίκτυου.

Για την επίλυση αυτού του προβλήματος, το Ethereum χρησιμοποιεί το πρωτόκολλο GHOST (Greedy Heaviest Observed Subtree), το οποίο αντιμετωπίζει την πρόκληση των παρωχημένων μπλοκ (stale blocks), τα οποία είναι μπλοκ που εξορύσσονται επιτυχώς αλλά τελικά απορρίπτονται επειδή επικράτησε άλλη μακρύτερη αλυσίδα στο δίκτυο. Στο GHOST, τα ανταγωνιστικά μπλοκ που αποκαλούνται «θείοι» (uncles) συνεισφέρουν στο συνολικό «βάρος» της αλυσίδας, αντί να απορρίπτονται πλήρως. Οι εξορύκτες που δημιουργούν τα μπλοκ θείων ανταμείβονται με το 87,5% της κανονικής ανταμοιβής ενός μπλοκ. Επιπλέον, όταν ένας εξορύκτης περιλαμβάνει ένα θείο στο μπλοκ του, λαμβάνει επιπλέον 3,125% ανταμοιβή, γεγονός που αυξάνει το βάρος της αλυσίδας του.

Στο Ethereum, δεν επικρατεί πάντα η μακρύτερη αλυσίδα, αλλά αυτή με το μεγαλύτερο «βάρος». Ο συνδυασμός του πρωτοκόλλου GHOST με την απόδειξη εργασίας, διασφαλίζει ότι η «βαρύτερη» αλυσίδα θα γίνει τελικά η κύρια αλυσίδα του δίκτυου (Σχήμα 11.6). Για να είναι οριστική μια συναλλαγή στο Ethereum, οι χρήστες πρέπει να περιμένουν τη δημιουργία περίπου 12 μπλοκ (το μπλοκ που περιέχει τη συναλλαγή + 11 μπλοκ επιβεβαίωσης).

Το επικρατόν μπλοκ (winning block) είναι το μπλοκ που αποτελεί μέρος της κύριας αλυσίδας και έχει επιλεγεί ως το έγκυρο μπλοκ που θα συνεχίσει να επεκτείνεται με τα επόμενα μπλοκ. Όπως απεικονίζεται στο Σχήμα 11.6, κατά τη διαδικασία δημιουργίας μπλοκ, ενδέχεται να δημιουργηθούν πολλά ανταγωνιστικά μπλοκ σχεδόν ταυτόχρονα, τα οποία βρίσκονται σε διαφορετικά παρακλάδια της αλυσίδας. Ωστόσο, το δίκτυο πρέπει να αποφασίσει ποιο από αυτά τα μπλοκ θα θεωρηθεί το «επικρατόν» ή κύριο μπλοκ.

Το επικρατόν μπλοκ επιλέγεται με βάση την αλυσίδα που έχει το μεγαλύτερο «βάρος» (σύμφωνα με το πρωτόκολλο GHOST) ή το μεγαλύτερο αριθμό επιβεβαιώσεων, ανάλογα με τον αλγόριθμο συναίνεσης. Έτσι, το επικρατόν μπλοκ γίνεται μέρος της αλυσίδας, ενώ τα ανταγωνιστικά μπλοκ είτε απορρίπτονται είτε θεωρούνται «θείοι» μπλοκ, που δεν είναι μέρος της κύριας αλυσίδας αλλά συμβάλλουν στο συνολικό βάρος της.



Σχήμα 11.6: Επισκόπηση του τρόπου λειτουργίας του Ethereum.

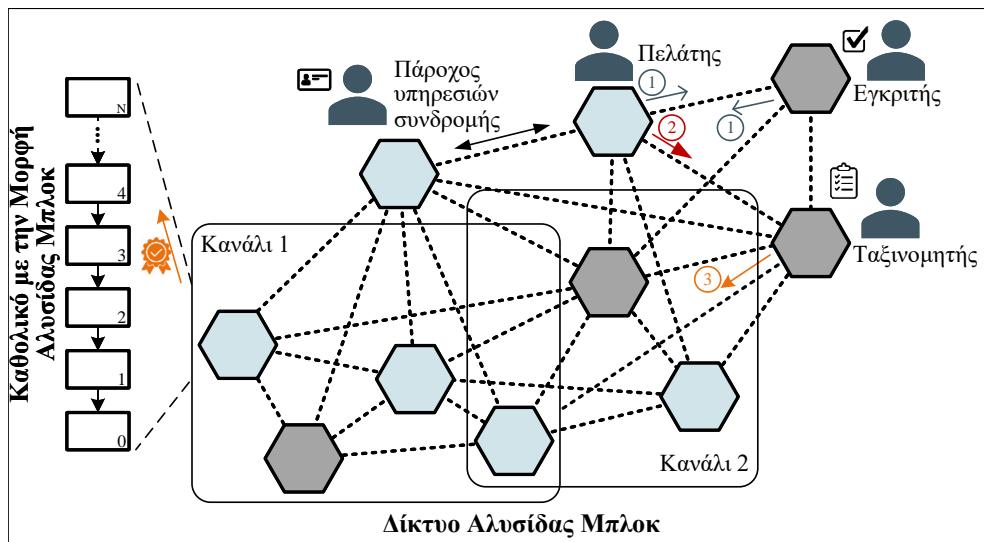
11.5.3 Hyperledger Fabric

Το Hyperledger Fabric [7] είναι ένα ανοιχτό κώδικα πλαίσιο που υποστηρίζεται από το Linux Foundation και προορίζεται ως βάση για την ανάπτυξη εφαρμογών αλυσίδας μπλοκ με αρθρωτή αρχιτεκτονική (modular architecture). Τα δεδομένα σε αυτή την αλυσίδα μπλοκ μπορούν να αποθηκεύονται σε διάφορες μορφές και

μπορούν να χρησιμοποιηθούν διάφοροι αλγόριθμοι συναίνεσης.

Το Hyperledger Fabric αποτελεί ένα είδος αλυσίδας μπλοκ με άδεια που συντηρείται από κάποια κοινοπράξια, όπου τα μέλη του δικτύου μπορούν να εγγραφούν μόνο μέσω ενός αξιόπιστου παρόχου συνδρομητικών υπηρεσιών (membership service provider) (Σχήμα 11.7). Όλοι οι συμμετέχοντες στο δίκτυο έχουν γνωστές ταυτότητες.

Το Hyperledger Fabric ορίζει την έννοια του καναλιού για την αντιμετώπιση περιπτώσεων όπου η ιδιωτικότητα και η εμπιστευτικότητα είναι σημαντικές και ο μειωμένος βαθμός διαφάνειας είναι αποδεκτός. Ένα κανάλι επιτρέπει σε μια ομάδα συμμετεχόντων να δημιουργήσουν ένα ξεχωριστό καθολικό συναλλαγών διαφυλάσσοντας έτσι την εμπιστευτικότητα των συναλλαγών τους από τα υπόλοιπα μέλη. Για παράδειγμα, μια εταιρεία μπορεί να δημιουργήσει ένα κανάλι με έναν ή κάποιους από τους πελάτες της, ώστε οι ανταγωνιστές της να μη μπορούν να δουν τις συναλλαγές στο καθολικό.



Σχήμα 11.7: Επισκόπηση του συστήματος Hyperledger Fabric.

Το καθολικό που σχετίζεται με ένα κανάλι περιλαμβάνει την παγκόσμια κατάσταση (world state) η οποία δείχνει την τελευταία κατάσταση του περιεχομένου του καθολικού, καθώς και το αρχείο καταγραφής συναλλαγών (transaction log) το οποίο καταγράφει όλο το ιστορικό των συναλλαγών που έχουν οδηγήσει στην τρέχουσα παγκόσμια κατάσταση. Το καθολικό ενός καναλιού περιέχει, επίσης, ένα μπλοκ ρυθμίσεων που καθορίζει πληροφορίες όπως πολιτικές και λίστες ελέγχου πρόσβασης.

Όπως και στο Ethereum, το Hyperledger Fabric υποστηρίζει έξυπνα συμβόλαια, τα οποία ονομάζονται κώδικες αλυσίδας (chaincodes), η εκτέλεση των οποίων βασίζεται στην παγκόσμια κατάσταση που αποθηκεύεται στο καθολικό για ένα κανάλι.

Στο Hyperledger Fabric συναντάμε τρεις διαφορετικούς τύπους κόμβων (Σχήμα 11.7):

- Κόμβοι πελάτες (client node):** Ένας κόμβος πελάτης (client) ενεργεί για λογαριασμό ενός τελικού χρήστη και συνδέεται με τους ομότιμους κόμβους για να εξασφαλίσει επικοινωνία με την αλυσίδα μπλοκ. Μπορεί να δημιουργήσει συναλλαγές και να μεταδίδει μηνύματα σε ταξινομητές κόμβους μέσω των καναλιών επικοινωνίας.
- Ταξινομητές κόμβοι (orderer node):** Ένας ταξινομητής (orderer) κόμβος επικυρώνει τις συναλλαγές με βάση την πολιτική εγκρίσεων και τις ταξινομεί ακολουθιακά πριν τις μεταδώσει στο δίκτυο. Η υπηρεσία ταξινόμησης που παρέχεται από τους ταξινομητές μπορεί να υποστηρίξει πολλαπλά κανάλια.
- Ομότιμοι κόμβοι (peer node):** Ένας ομότιμος (peer) κόμβος λαμβάνει ενημερώσεις κατάστασης με

τη μορφή συναλλαγών από τους ταξινομητές, πραγματοποιεί συναλλαγές και διατηρεί την κατάσταση του καθολικού.

4. **Εγκριτές κόμβοι (endorser node):** Μερικοί ομότιμοι κόμβοι μπορούν να αναλάβουν έναν ειδικό ρόλο, αυτόν του εγκριτή (endorser) που ελέγχει και εγκρίνει συναλλαγές βάσει της πολιτικής που εφαρμόζει ο σχετικός κώδικας αλυσίδας.

Οι έλεγχοι, που πραγματοποιούνται κατά τη διάρκεια του κύκλου ζωής μιας συναλλαγής, μπορούν να χωριστούν σε τρία στάδια:

1. Την **έγκριση**, η οποία καθοδηγείται από την πολιτική που καθορίζει ποιοι ομότιμοι κόμβοι εγκρίνουν μια συγκεκριμένη συναλλαγή.
2. Την **ταξινόμηση**, η οποία δέχεται τις εγκεκριμένες συναλλαγές και τις ταξινομεί σε μια ακολουθία που θα εισαχθεί στο αντίστοιχο καθολικό με την αποστολή των μπλοκ σε όλους τους ομότιμους κόμβους του καναλιού.
3. Την **επικύρωση**, η οποία ελέγχει την ορθότητα ενός συνόλου ταξινομημένων συναλλαγών σε ένα μπλοκ, λαμβάνοντας υπόψη την πολιτική εγκρίσεων και τους ελέγχους εκδόσεων για την ακεραιότητα των δεδομένων.

Έτσι, όπως αποτυπώνεται στο Σχήμα 11.7, η ροή μιας συναλλαγής ξεκινά όταν: ένας κόμβος πελάτης υποβάλει ένα αίτημα συναλλαγής σε έναν ή περισσότερους εγκριτές κόμβους (1). Οι εγκριτές κόμβοι εκτελούν την προτεινόμενη συναλλαγή προσομοιωτικά, χωρίς να την καταγράψουν στο καθολικό, και δημιουργούν μια υπογεγραμμένη έγκριση (endorsement), την οποία επιστρέφουν στον πελάτη (2). Αφού ο πελάτης λάβει τις απαραίτητες εγκρίσεις, στέλνει τη συναλλαγή στους ταξινομητές κόμβους (3), οι οποίοι αναλαμβάνουν τη δέσμη των συναλλαγών (batching) και τη δημιουργία μπλοκ με βάση τη χρονολογική σειρά. Το νέο μπλοκ αποστέλλεται στους Ομότιμους κόμβους (4), οι οποίοι επικυρώνουν τις εγκρίσεις και την πολιτική συναίνεσης και στη συνέχεια ενημερώνουν το καθολικό (5) προσθέτοντας το μπλοκ. Οι ομότιμοι κόμβοι διανέμουν το ενημερωμένο καθολικό στο δίκτυο, ολοκληρώνοντας έτσι τη διαδικασία της συναλλαγής.

Η αρθρωτή αρχιτεκτονική του Hyperledger Fabric επιτρέπει την επιλογή του κατάλληλου αλγορίθμου συναίνεσης για κάθε μια από τις τρεις φάσεις, έτσι ώστε οι εφαρμογές, ανάλογα με τις απαιτήσεις τους, να μπορούν να κάνουν χρήση διαφορετικών μοντέλων για έγκριση, ταξινόμηση και επικύρωση. Επιπλέον της έγκρισης, της ταξινόμησης και της επικύρωσης, κατά τη διαδικασία συναίνεσης πραγματοποιείται και η επαλήθευση ταυτότητας των συμμετεχόντων.

11.6 Προκλήσεις Ασφάλειας και Λειτουργίας των Αλυσίδων Μπλοκ

Αν και οι αλυσίδες μπλοκ προσφέρουν εξαιρετικές δυνατότητες για την ενίσχυση της ασφάλειας σε αποκεντρωμένα περιβάλλοντα, παρουσιάζουν επίσης προκλήσεις που αφορούν τόσο την ασφάλεια όσο και τη λειτουργία τους. Στην παρούσα ενότητα αναλύονται σημαντικά ζητήματα, όπως οι επιθέσεις ασφάλειας, η αποδοτικότητα του δικτύου και φαινόμενα όπως τα παρωχημένα και ορφανά μπλοκ, που επηρεάζουν τη συνολική λειτουργία των αλυσίδων μπλοκ.

11.6.1 Επίθεση Sybil

Η επίθεση Sybil είναι μια μέθοδος με την οποία ένας κακόβουλος χρήστης προσπαθεί να διαταράξει τη λειτουργία ενός δικτύου ομότιμων κόμβων δημιουργώντας πολλούς ψεύτικους κόμβους [8]. Αυτοί οι ψεύτικοι κόμβοι παρουσιάζονται ως νόμιμοι και αξιόπιστοι στους υπόλοιπους συμμετέχοντες του δικτύου.

Ο στόχος της επίθεσης είναι να αποκτήσει ο επιτιθέμενος τον έλεγχο ενός σημαντικού μέρους του δικτύου, ώστε να μπορεί να επιβεβαιώνει μη εξουσιοδοτημένες συναλλαγές ή να τροποποιεί έγκυρες συναλλαγές. Όσο

περισσότεροι κόμβοι Sybil υπάρχουν στο δίκτυο, τόσο αυξάνονται οι πιθανότητες για τον επιτιθέμενο να επιτύχει αυτό το στόχο. Ένας από τους κινδύνους που προκύπτουν από την επίθεση Sybil είναι η δυνατότητα πραγματοποίησης διπλής δαπάνης, όπου ο ίδιος επιτιθέμενος ξοδεύει τα ίδια κρυπτονομίσματα περισσότερες από μία φορές. Όταν ένας κακόβουλος χρήστης κατέχει έναν σημαντικό αριθμό κόμβων στο δίκτυο, αυξάνονται και οι πιθανότητες για μια τέτοια επίθεση.

11.6.2 Η Επίθεση του 51%

Η επίθεση του 51% είναι μια σοβαρή απειλή για τις αλυσίδες μπλοκ που χρησιμοποιούν κυρίως αλγόριθμους συναίνεσης Proof of Work (PoW). Αυτή η επίθεση συμβαίνει όταν ένας εξορύκτης ή μια ομάδα εξορυκτών αποκτήσει τον έλεγχο του 51% ή περισσότερου των κόμβων εξόρυξης στο δίκτυο και έτσι μπορεί να υπερισχύσει στην αλυσίδα μπλοκ. Με αυτό τον τρόπο, οι επιτιθέμενοι αποκτούν τη δυνατότητα να ελέγχουν το δίκτυο και να εκτελούν διάφορες κακόβουλες ενέργειες. Η επίθεση του 51% σχετίζεται με την επίθεση Sybil καθώς και οι δύο εκμεταλλεύονται τη δομή των αποκέντρωμένων δικτύων, αλλά με διαφορετικούς τρόπους. Στην επίθεση του 51%, ένας επιτιθέμενος αποκτά τον έλεγχο του 51% της υπολογιστικής ισχύος ή της ισχύος επικύρωσης στο δίκτυο, επιτρέποντάς του να αλλοιώσει την αλυσίδα μπλοκ, όπως να κάνει διπλές δαπάνες ή να εμποδίσει τη δημιουργία νέων μπλοκ. Στην επίθεση Sybil, ο επιτιθέμενος δημιουργεί πολλές ψεύτικες ταυτότητες ή κόμβους, προσπαθώντας να επηρεάσει τη διαδικασία συναίνεσης ή τη λειτουργία του δικτύου. Μια επίθεση Sybil μπορεί να βοηθήσει στην εκτέλεση μιας επίθεσης του 51% αν μέσω αυτών των ψεύτικων κόμβων ο επιτιθέμενος αποκτήσει την πλειοψηφία της ισχύος. Ωστόσο, η επίθεση του 51% επικεντρώνεται στον έλεγχο της υπολογιστικής ισχύος, ενώ η επίθεση Sybil αφορά τη δημιουργία πολλαπλών ταυτοτήτων έτσι ώστε ο επιτιθέμενος να επηρεάσει το δίκτυο.

Όταν οι επιτιθέμενοι ελέγχουν την πλειοψηφία της υπολογιστικής ισχύος του δικτύου, μπορούν να:

- Εμποδίσουν την επικύρωση νέων συναλλαγών απορρίπτοντας ή καθυστερώντας τις συναλλαγές που προέρχονται από άλλους εξορύκτες, και έτσι να αποτρέψουν την ένταξή τους στα μπλοκ.
- Αναιρέσουν τις δικές τους συναλλαγές και να ξοδέψουν τα ίδια κρυπτονομίσματα περισσότερες από μία φορές, δημιουργώντας την λεγόμενη επίθεση διπλής δαπάνης (Ενότητα 11.6.6).
- Εμποδίσουν άλλους εξορύκτες από το να δημιουργήσουν μπλοκ καθώς ελέγχουν το μεγαλύτερο μέρος της υπολογιστικής ισχύος και έτσι μπορούν να εξορύξουν νέα μπλοκ πιο γρήγορα από τους ανταγωνιστές τους, εξασφαλίζοντας ότι τα μπλοκ που θα προστεθούν στην αλυσίδα θα είναι τα δικά τους.

Οι επιπτώσεις μιας επίθεσης του 51% μπορεί να είναι καταστροφικές για το δίκτυο της αλυσίδας μπλοκ. Η εμπιστοσύνη στο δίκτυο μπορεί να διαταραχθεί, οι συναλλαγές να θεωρηθούν αναξιόπιστες και η αξία του κρυπτονομίσματος να μειωθεί δραματικά. Παρόλο που η πιθανότητα μιας τέτοιας επίθεσης είναι μικρή για μεγάλα δίκτυα με υψηλό αριθμό κόμβων εξόρυξης, όπως το Bitcoin, παραμένει μια σημαντική ανησυχία για μικρότερα και νεότερα δίκτυα που δεν έχουν ακόμα αρκετή υπολογιστική ισχύ για να αποτρέψουν τέτοιες επιθέσεις.

Για να αντιμετωπιστεί η απειλή της επίθεσης του 51%, οι αλυσίδες μπλοκ μπορούν να εφαρμόσουν διάφορες τεχνικές, όπως η διαφοροποίηση των αλγορίθμων συναίνεσης ή η ενίσχυση της αποκέντρωσης του δικτύου, ώστε να δυσχεράνουν τον έλεγχο της πλειοψηφίας από έναν μεμονωμένο εξορύκτη ή ομάδα εξορυκτών.

11.6.3 Εγωιστική Εξόρυξη

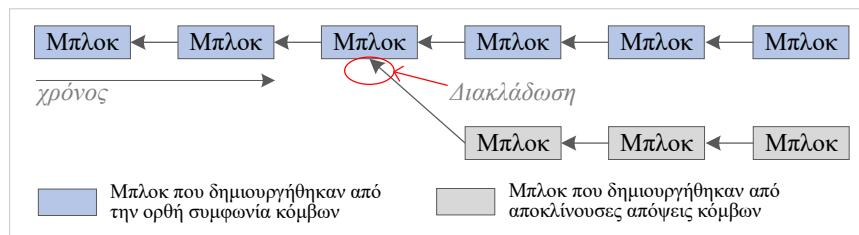
Η εγωιστική εξόρυξη (selfish mining), είναι μια τακτική που μπορεί να χρησιμοποιηθεί από κακόβουλους κόμβους εξόρυξης σε δίκτυα αλυσίδων μπλοκ χωρίς άδεια [9]. Σε αυτή την τακτική, οι κόμβοι εξόρυξης δεν δημοσιεύουν άμεσα τα μπλοκ που εξορύσσουν στο δίκτυο. Αντίθετα, κρατούν τα μπλοκ κρυφά και συνεχίζουν την εξόρυξη για να δημιουργήσουν μια μεγαλύτερη ιδιωτική αλυσίδα από την υπάρχουσα δημόσια αλυσίδα. Ο

σκοπός αυτής της στρατηγικής είναι να αποκτήσουν οι κακόβουλοι εξορύκτες ένα πλεονέκτημα, αυξάνοντας τα κέρδη τους όταν τελικά δημοσιεύσουν τα κρυφά μπλοκ στο δίκτυο.

Αυτή η τακτική μπορεί να έχει αρνητικές συνέπειες για τους υπόλοιπους κόμβους του δικτύου, καθώς μπορεί να ακυρωθούν έγκυρα μπλοκ που εξορύχθηκαν από έντιμους κόμβους. Επιπλέον, αν δύο ομάδες κακόβουλων εξορυκτών ανταγωνίζονται μεταξύ τους για να προσθέσουν τα δικά τους μπλοκ στο δίκτυο, μπορεί να δημιουργηθούν διακλαδώσεις στην αλυσίδα μπλοκ, οδηγώντας σε αστάθεια και καθυστερήσεις στη συναίνεση του δικτύου.

11.6.4 Διακλάδωση Αλυσίδας Μπλοκ

Μια διακλάδωση αλυσίδας μπλοκ (blockchain fork) [3], είναι μια κατάσταση στην οποία το δίκτυο χωρίζεται σε δύο ή περισσότερες αλυσίδες λόγω διαφορών στην κατάσταση του καθολικού (Σχήμα 11.8). Αυτές οι διακλαδώσεις μπορούν να προκύψουν είτε ακούσια είτε σκόπιμα.



Σχήμα 11.8: Δημιουργία διακλαδώσης σε μια αλυσίδα μπλοκ.

Ακούσιες διακλαδώσεις μπορεί να συμβούν εξαιτίας σφαλμάτων στο πρωτόκολλο ή ασυμβατοτήτων σε αναβαθμίσεις λογισμικού των πελατών. Σε αυτές τις περιπτώσεις, οι κόμβοι του δικτύου μπορεί να έχουν διαφορετικές απόψεις για το ποια είναι η τρέχουσα αλυσίδα μπλοκ, οδηγώντας σε προσωρινές ή ακόμα και μόνιμες διαφορές στην κατάσταση του καθολικού.

Σκόπιμες διακλαδώσεις μπορεί να προκληθούν από κακόβουλες ενέργειες, όπως η χρήση κόμβων Sybil που ακολουθούν διαφορετικούς κανόνες επικύρωσης ή από εγωιστική εξόρυξη. Επιπλέον, διακλαδώσεις μπορούν να προκύψουν από προτεινόμενες αλλαγές στο λογισμικό της αλυσίδας μπλοκ, που μπορεί να οδηγήσουν είτε σε «μαλακές» (soft) είτε σε «σκληρές» (hard) διακλαδώσεις. Οι διακλαδώσεις μπορούν επίσης να προκαλέσουν καθυστερήσεις στη συναίνεση του δικτύου, οι οποίες μπορεί να οδηγήσουν σε άλλες επιθέσεις, όπως η διπλή δαπάνη.

Μια «μαλακή» διακλαδωση είναι μια αναβάθμιση λογισμικού που είναι συμβατή με τις προηγούμενες εκδόσεις. Αυτό σημαίνει ότι οι χρήστες που χρησιμοποιούν παλαιότερες εκδόσεις του λογισμικού θα εξακολουθήσουν να αναγνωρίζουν τα μπλοκ που δημιουργούνται από τις νεότερες εκδόσεις και αντίστροφα. Ονομάζεται «μαλακή» διότι και οι δύο ομάδες χρηστών συνεχίζουν να εργάζονται στην ίδια αλυσίδα.

Αντίθετα, μια «σκληρή» διακλαδωση είναι μια αναβάθμιση λογισμικού που δεν είναι συμβατή με τις προηγούμενες εκδόσεις. Αυτό σημαίνει ότι οι χρήστες που εκτελούν παλαιότερες εκδόσεις του λογισμικού δεν θα αναγνωρίζουν τα μπλοκ που δημιουργούνται από τις νεότερες εκδόσεις και το αντίστροφο. Οι «σκληρές» διακλαδώσεις μπορεί να είναι ιδιαίτερα προβληματικές, καθώς μπορούν να οδηγήσουν σε μόνιμη διάσπαση της αλυσίδας μπλοκ και σε διπλές δαπάνες.

11.6.5 Καθυστέρηση Συναίνεσης

Η καθυστέρηση συναίνεσης (consensus delay) [3] είναι μια κατάσταση που μπορεί να επηρεάσει την αποτελεσματικότητα και την ασφάλεια ενός δικτύου αλυσίδας μπλοκ. Αυτή η καθυστέρηση προκύπτει όταν οι κόμβοι του δικτύου δυσκολεύονται να συμφωνήσουν σε μια ενιαία κατάσταση της αλυσίδας μπλοκ, οδηγώντας σε επιβράδυνση της επικύρωσης των συναλλαγών και της δημιουργίας νέων μπλοκ. Το πρόβλημα αυτό

είναι ιδιαίτερα κρίσιμο για εφαρμογές όπου ο χρόνος είναι ζωτικής σημασίας και η συναίνεση πρέπει να επιτευχθεί όσο το δυνατόν συντομότερα.

Οι αιτίες της καθυστέρησης συναίνεσης μπορεί να περιλαμβάνουν:

- Δίκτυο υψηλής συμφόρησης: Όταν το δίκτυο είναι υπερφορτωμένο με πολλές συναλλαγές, οι κόμβοι μπορεί να χρειαστούν περισσότερο χρόνο για να επεξεργαστούν και να επικυρώσουν κάθε συναλλαγή.
- Ασυμβατότητες λογισμικού: Οι διαφορές στις εκδόσεις του λογισμικού που χρησιμοποιούν οι κόμβοι μπορεί να προκαλέσουν καθυστερήσεις στη διαδικασία συναίνεσης, καθώς οι κόμβοι πρέπει να ενημερώσουν τα πρωτόκολλά τους για να επιτύχουν ομοφωνία.
- Κακόβουλες ενέργειες: Επιθέσεις όπως η επίθεση Sybil ή εγωιστική εξόρυξη μπορούν να προκαλέσουν καθυστερήσεις στη συναίνεση, καθώς οι κακόβουλοι κόμβοι προσπαθούν να διαταράξουν τη διαδικασία επικύρωσης.

Η καθυστέρηση συναίνεσης μπορεί να έχει σημαντικές επιπτώσεις στο δίκτυο της αλυσίδας μπλοκ, όπως είναι η αύξηση του κινδύνου επιθέσεων και η δυσλειτουργία του δικτύου. Οι επιθέσεις διπλής δαπάνης γίνονται πιο πιθανές όταν υπάρχει καθυστέρηση στη συναίνεση, καθώς οι επιτιθέμενοι έχουν περισσότερο χρόνο για να εκτελέσουν κακόβουλες ενέργειες. Επιπλέον, οι συνεχείς καθυστερήσεις μπορεί να οδηγήσουν σε διακλαδώσεις και άλλες ασυμβατότητες στο δίκτυο, καθιστώντας το σύστημα λιγότερο αποδοτικό και αξιόπιστο.

Η καθυστέρηση συναίνεσης είναι ένα σημαντικό ζήτημα για τις αλυσίδες μπλοκ και απαιτεί συνεχή παρακολούθηση και βελτιώσεις για να διασφαλιστεί η αποδοτική και ασφαλής λειτουργία του δικτύου. Για να μειωθεί η καθυστέρηση συναίνεσης, μπορούν να εφαρμοστούν διάφορες τεχνικές:

- Βελτιστοποίηση του πρωτοκόλλου: Αναβαθμίσεις και βελτιώσεις στο πρωτόκολλο συναίνεσης μπορούν να αυξήσουν την ταχύτητα και την αποδοτικότητα της διαδικασίας συναίνεσης.
- Αποφυγή συμφόρησης δικτύου: Μέτρα όπως η αύξηση του μεγέθους των μπλοκ ή η βελτιστοποίηση της διαχείρισης των συναλλαγών μπορούν να μειώσουν την υπερφόρτωση του δικτύου.
- Αυστηρότερος έλεγχος κακόβουλων ενεργειών: Η ανίχνευση και η αποτροπή κακόβουλων ενεργειών μπορούν να συμβάλουν στη διατήρηση της ομαλής λειτουργίας του δικτύου και στη μείωση των καθυστερήσεων στη συναίνεση.

11.6.6 Διπλή Δαπάνη

Η διπλή δαπάνη (double spending) είναι ένα σοβαρό ζήτημα που αφορά την ακεραιότητα των συναλλαγών σε ένα δίκτυο αλυσίδας μπλοκ [10]. Η διπλή δαπάνη συμβαίνει όταν ένας χρήστης προσπαθεί να ξοδέψει τα ίδια κρυπτονομίσματα περισσότερες από μία φορές. Σε ένα παραδοσιακό χρηματοπιστωτικό σύστημα, η κεντρική αρχή επαληθεύει τις συναλλαγές και αποτρέπει τη διπλή δαπάνη. Ωστόσο, σε ένα αποκεντρωμένο δίκτυο αλυσίδας μπλοκ, αυτή η επαλήθευση πρέπει να γίνεται από το ίδιο το δίκτυο, χωρίς την ύπαρξη κεντρικής αρχής.

Στις αλυσίδες μπλοκ, η άμεση ενημέρωση μεταξύ των κόμβων του δικτύου είναι απαραίτητη για την αποφυγή της διπλής δαπάνης. Όπως έχει ήδη αναφερθεί, όταν ένας χρήστης δημιουργεί μια συναλλαγή, αυτή μεταδίδεται στο δίκτυο. Οι υπόλοιποι κόμβοι του δικτύου επαληθεύουν τη συναλλαγή και, εφόσον είναι έγκυρη, την συμπεριλαμβάνουν σε μια προσωρινή λίστα συναλλαγών από την οποία ένας εξορύκτης επιλέγει αυτές που θα συμπεριλάβει σε ένα νέο μπλοκ. Μόλις το μπλοκ που περιέχει τη συναλλαγή προστεθεί στην αλυσίδα, η συναλλαγή θεωρείται επιβεβαιωμένη και τα κρυπτονομίσματα δεν μπορούν να δαπανηθούν ξανά.

Παρά τη διαδικασία αυτή, η διπλή δαπάνη μπορεί να συμβεί μέσω διάφορων τεχνικών. Μια κοινή μέθοδος είναι η επίθεση ανταγωνισμού (race attack), όπου ένας χρήστης στέλνει δύο αντικρουόμενες συναλλαγές σε διαφορετικούς κόμβους του δικτύου σχεδόν ταυτόχρονα. Εάν οι κόμβοι δεν επικοινωνήσουν επαρκώς μεταξύ

τους, μπορεί να επιβεβαιώσουν διαφορετικές συναλλαγές, οδηγώντας σε διπλή δαπάνη. Μια άλλη μέθοδος είναι η επίθεση Finney, όπου ένας εξόρυκτης δημιουργεί ένα μπλοκ που περιέχει μια συναλλαγή και το κρατάει κρυφό, ενώ πραγματοποιεί μια δεύτερη συναλλαγή με τα ίδια κρυπτονομίσματα. Αν η δεύτερη συναλλαγή επιβεβαιωθεί από το δίκτυο πριν δημοσιοποιηθεί το κρυφό μπλοκ, μπορεί να δημιουργηθεί διπλή δαπάνη. Μια ακόμη μορφή επίθεσης διπλής δαπάνης είναι η επίθεση του 51% όπου οι επιτιθέμενοι μπορούν να επιβεβαιώσουν μόνο τις συναλλαγές που επιθυμούν και να αναιρέσουν τις δικές τους συναλλαγές, επιτρέποντας έτσι τη διπλή δαπάνη.

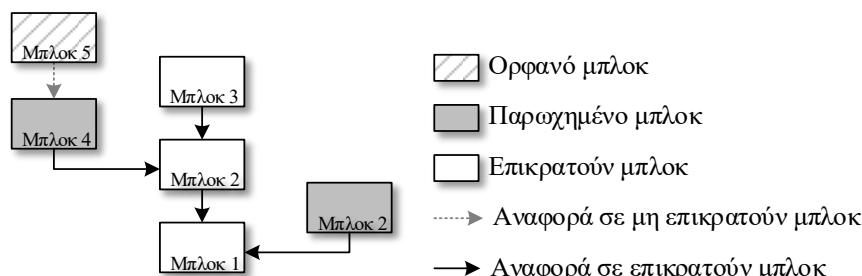
Για την αντιμετώπιση της διπλής δαπάνης, οι αλυσίδες μπλοκ εφαρμόζουν διάφορες τεχνικές ασφαλείας. Η επιβεβαίωση πολλών μπλοκ πριν θεωρηθεί μια συναλλαγή οριστική είναι μία από αυτές τις τεχνικές. Επίσης, η χρήση εναλλακτικών αλγορίθμων συναίνεσης, όπως το Proof of Stake (PoS), μπορεί να μειώσει την πιθανότητα επίθεσης του 51% και να ενισχύσει την ασφάλεια του δικτύου. Επιπλέον, η συνεχής παρακολούθηση και βελτίωση των πρωτοκόλλων ασφαλείας συμβάλλει στην προστασία των αλυσίδων μπλοκ από τη διπλή δαπάνη.

11.6.7 Παρωχημένα και Ορφανά Μπλοκ

Στο πλαίσιο των αλυσίδων μπλοκ, μπορούν να προκύψουν δύο τύποι μπλοκ που δεν ενσωματώνονται στην κύρια αλυσίδα (Σχήμα 11.9): τα παρωχημένα μπλοκ και τα ορφανά μπλοκ.

Τα παρωχημένα μπλοκ (stale block) είναι μπλοκ που έχουν εξορυχθεί επιτυχώς, αλλά δεν ενσωματώνονται στην κύρια αλυσίδα επειδή δημιουργήθηκε μια μεγαλύτερη αλυσίδα που δεν τα περιλαμβάνει. Αυτό μπορεί να συμβεί λόγω ανταγωνισμού μεταξύ των εξορυκτών ή εξαιτίας επιθέσεων εγωιστικής εξόρυξης. Όταν δύο εξορύκτες δημιουργούν μπλοκ σχεδόν ταυτόχρονα, μόνο ένα από αυτά θα ενσωματωθεί στην κύρια αλυσίδα, ενώ το άλλο θα γίνει παρωχημένο, καθώς το δίκτυο τυπικά επιλέγει την πιο μακρά αλυσίδα ως έγκυρη.

Τα ορφανά μπλοκ (orphan block) είναι μπλοκ των οποίων η επικεφαλίδα περιέχει μια σύνοψη του προηγούμενου μπλοκ που δεν αντιστοιχεί σε κάποιο έγκυρο μπλοκ της αλυσίδας [11]. Αυτά τα μπλοκ δεν μπορούν να επικυρωθούν και να ενσωματωθούν στην αλυσίδα επειδή το προηγούμενο μπλοκ τους είτε δεν υπάρχει είτε δεν είναι έγκυρο. Τα ορφανά μπλοκ δημιουργούνται συχνά λόγω καθυστερήσεων στη μετάδοση των μπλοκ στο δίκτυο ή λόγω δυσλειτουργιών στο πρωτόκολλο.



Σχήμα 11.9: Παράδειγμα παρωχημένων και ορφανών μπλοκ.

Βιβλιογραφία

- [1] Satoshi Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. 2008.
- [2] Vitalik Buterin. “Ethereum: A next-generation smart contract and decentralized application platform”. In: *Ethereum White Paper*. Vol. 3. 37. Ethereum Foundation. 2014, pp. 2–1.
- [3] Sachin Shetty, Charles A. Kamhoua, and Laurent L. Njilla. *Blockchain for distributed systems security*. Hoboken: John Wiley IEEE computer society, 2019. ISBN: 978-1-119-51962-1.

- [4] Miguel Castro and Barbara Liskov. “Practical Byzantine Fault Tolerance”. In: *Proceedings of the Third Symposium on Operating Systems Design and Implementation (OSDI)*. New Orleans, Louisiana, USA: USENIX Association, 1999, pp. 173–186. URL: <https://www.usenix.org/conference/osdi-99/practical-byzantine-fault-tolerance>.
- [5] Satoshi Nakamoto. “Bitcoin: A peer-to-peer electronic cash system”. In: *Decentralized Business Review* (2008).
- [6] Gavin Wood. “Ethereum: A secure decentralised generalised transaction ledger”. In: *Ethereum project yellow paper 151* (2014), pp. 1–32.
- [7] Elli Androulaki et al. “Hyperledger fabric: a distributed operating system for permissioned blockchains”. In: *Proceedings of the Thirteenth EuroSys Conference*. 2018, pp. 1–15.
- [8] John R. Douceur. “The Sybil Attack”. In: *Peer-to-Peer Systems*. Ed. by Peter Druschel, Frans Kaashoek, and Antony Rowstron. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 251–260. ISBN: 978-3-540-45748-0.
- [9] Ittay Eyal and Emin Gün Sirer. “Majority is not enough: bitcoin mining is vulnerable”. In: *Commun. ACM* 61.7 (June 2018), pp. 95–102. ISSN: 0001-0782. DOI: [10.1145/3212998](https://doi.org/10.1145/3212998).
- [10] Ghassan O. Karame, Elli Androulaki, and Srdjan Capkun. “Double-spending fast payments in bitcoin”. In: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*. CCS ’12. Raleigh, North Carolina, USA: Association for Computing Machinery, 2012, pp. 906–917. ISBN: 9781450316514. DOI: [10.1145/2382196.2382292](https://doi.org/10.1145/2382196.2382292).
- [11] Christian Decker and Roger Wattenhofer. “Information propagation in the Bitcoin network”. In: *IEEE P2P 2013 Proceedings*. IEEE. 2013, pp. 1–10.

ΚΕΦΑΛΑΙΟ 12

ΔΙΑΣΦΑΛΙΣΗ ΑΠΟΡΡΗΤΟΥ ΚΑΙ ΙΔΙΩΤΙΚΟΤΗΤΑΣ

Περίληψη

Τα τελευταία χρόνια αναδυόμενες τεχνολογίες έρχονται να συμβάλλουν στη δημιουργία ολοκληρωμένων εφαρμογών με απότερο σκοπό την προστασία της ιδιωτικότητας των πληροφοριών και ειδικά σε ότι αφορά τη διαρροή ευαίσθητων πληροφοριών που προκύπτουν από την ανάλυση δεδομένων. Ανάλογα με την φύση αυτών των δεδομένων, προσωπικών ή εταιρικών δεδομένων, μπορεί να τίθενται ζητήματα ιδιωτικότητας και εμπορικού/εταιρικού απορρήτου, αντίστοιχα. Στόχος αυτού του κεφαλαίου είναι να παρουσιάσει παραδείγματα τέτοιων εφαρμογών διασφάλισης της ιδιωτικότητας σε διάφορες ερευνητικές περιοχές. Πιο αναλυτικά, στην Ενότητα 12.1 γίνεται μια εισαγωγή αναφορικά με το απόρρητο και την ιδιωτικότητα, προσπαθώντας να αποσαφηνιστούν αυτές οι δύο έννοιες όσον αφορά τις ομοιότητες και τις διαφορές που παρουσιάζουν. Στην συνέχεια, η Ενότητα 12.2 επικεντρώνεται και παρουσιάζει γνωστά δίκτυα ανωνυμίας (Mix-Nets, TOR και I2P). Ακολουθεί, η Ενότητα 12.3 όπου οργανώνει και παρουσιάζει συστήματα ηλεκτρονικής ψηφοφορίας με βάση τις κρυπτογραφικές τεχνικές που χρησιμοποιούνται για τη διασφάλιση της ιδιωτικότητας. Στην Ενότητα 12.4 παρουσιάζονται διάφορες λύσεις ιδιωτικότητας για την πραγματοποίηση ηλεκτρονικών δημοπρασιών, ενώ στην Ενότητα 12.5 συνοψίζονται διάφορα σχήματα ιδιωτικής ανάκτησης πληροφοριών από βάσεις δεδομένων. Τέλος, στις Ενότητες 12.6 και 12.7 παρουσιάζονται λύσεις που διασφαλίζουν την ιδιωτικότητα στην εξόρυξη δεδομένων και στην μηχανική μάθηση, αντίστοιχα, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών αρχών της συμμετρικής κρυπτογράφησης (Κεφάλαιο 1) και της κρυπτογράφησης δημοσίου κλειδιού (Κεφάλαιο 2), το τρόπο λειτουργίας των αλυσίδων μπλοκ (Κεφάλαιο 11), κάποιων βασικών εννοιών της μετα-κβαντικής κρυπτογραφίας (Κεφάλαιο 15), και κυρίως των μηχανισμών ενίσχυσης του απορρήτου (Κεφάλαιο 8).

12.1 Απόρρητο και Ιδιωτικότητα

Οι όροι «απόρρητο» (confidentiality) και «ιδιωτικότητα» (privacy) πολύ συχνά συγχέονται ή θεωρούνται ισοδύναμοι, αλλά στην πραγματικότητα αντιπροσωπεύουν δύο εντελώς διαφορετικές έννοιες. Ενώ αυτοί οι όροι πολλές φορές χρησιμοποιούνται εναλλάξ, στην πραγματικότητα αναφέρονται σε ξεχωριστές αλλά συναφείς έννοιες. Ο ακριβής ορισμός αυτών των δύο όρων συχνά εξαρτάται από το πλαίσιο στο οποίο αναφέρονται. Σε ορισμένους τομείς μπορούν επίσης να έχουν πολύ συγκεκριμένη ερμηνεία, όπως όταν χρησιμοποιούνται στον ιατρικό και στον νομικό τομέα. Το απόρρητο και η ιδιωτικότητα επηρεάζουν διαφορετικά τα δεδομένα που παρέχονται ή συλλέγονται καθημερινά για ένα άτομο. Για να καταστεί σαφής η διαφορά μεταξύ του απόρρητου και της ιδιωτικότητας, με απλά λόγια, το απόρρητο αφορά τη μυστικότητα των δεδομένων γενικώς, ενώ η ιδιωτικότητα αφορά την περιορισμένη πρόσβαση σε δεδομένα που σχετίζονται με ένα συγκεκριμένο άτομο.

Απόρρητο Δεδομένων: Το απόρρητο των δεδομένων θεωρείται συχνά το ίδιο με την προστασία της εμπιστευτικότητας των δεδομένων. Άλλα πέρα από τις θεμελιώδεις και οργανωτικές δικλείδες ασφαλείας, το απόρρητο επικεντρώνεται στον τρόπο διαβάθμισης και διαχείρισης της πρόσβασης σε δεδομένα. Αυτό περιλαμβάνει τόσο ψηφιακές όσο και φυσικές μορφές, εφαρμόζοντας μέτρα προστασίας και περιορισμών που ταιριάζουν στο επίπεδο ευαισθησίας των δεδομένων. Δηλαδή, αναγνωρίζει ότι όλα τα δεδομένα δεν είναι ίδια και κατ' επέκταση δεν εφαρμόζει σε όλα τα δεδομένα το ίδιο επίπεδο προστασίας. Τα επίπεδα ευαισθησίας μπορούν και πρέπει να ορίζονται με διαφορετικούς τρόπους ώστε να ταιριάζουν στο πλαίσιο στο οποίο εφαρμόζονται. Αυτό που είναι σημαντικό είναι ότι αντικατοπτρίζει τη φύση των δεδομένων που συλλέγονται, υποβάλλονται σε επεξεργασία και αποθηκεύονται, καθώς και πως αυτά τα δεδομένα πρέπει να αντιμετωπίζονται ανάλογα με την ευαισθησία τους. Για παράδειγμα, ακολουθούν τρία παραδείγματα δεδομένων με διαφορετικά επίπεδα ευαισθησίας:

- **Εταιρικά εμπιστευτικά δεδομένα:** Ο χάρτης εξέλιξης ενός προϊόντος, η οικονομική απόδοση και η εσωτερική τεκμηρίωση των διαδικασιών μπορούν να κοινοποιηθούν σε όλους εντός του οργανισμού αλλά όχι εκτός αυτού.
- **Περιορισμένα (restricted) δεδομένα:** Τα πρακτικά συνεδρίασης του διοικητικού συμβουλίου, η εταιρική στρατηγική και η τεκμηρίωση του επιχειρηματικού σχεδιασμού ενδέχεται να περιορίζονται συνήθως μόνο στην εκτελεστική ηγετική ομάδα ενός οργανισμού.
- **Διαβαθμισμένα (classified) δεδομένα:** Δεδομένα που εγκυμονούν αυξημένους κινδύνους ασφάλειας ή επαναλαμβανόμενους κινδύνους ενδέχεται να είναι εξαιρετικά περιορισμένα σε ελάχιστα άτομα ενός οργανισμού.

Η πρόσβαση σε οποιαδήποτε μη δημόσια δεδομένα πρέπει να ακολουθεί την αρχή των ελαχίστων προνομίων. Δηλαδή θα πρέπει να παρέχεται πρόσβαση μόνο σε αυτούς που τα χρειάζονται για μια νόμιμη επιχειρησιακή ανάγκη. Ωστόσο, όπως βλέπουμε στα παραπάνω παραδείγματα, το απόρρητο των δεδομένων παρέχεται σε διαφορετικές μορφές και διαφορετικά επίπεδα ευαισθησίας που πρέπει επίσης να ληφθούν υπόψη στον τρόπο χειρισμού αυτών των δεδομένων και στο επίπεδο αυστηρότητας που εφαρμόζεται για την αποτροπή της πρόσβασης σε αυτά από μη εξουσιοδοτημένα άτομα.

Ιδιωτικότητα Δεδομένων: Η ιδιωτικότητα των δεδομένων συνδέεται στενά με το απόρρητο, αλλά προσεγγίζει τα δεδομένα από διαφορετική οπτική γωνία. Η ιδιωτικότητα ορίζει ότι το άτομο που καθορίζει τι είναι κατάλληλο να πραγματοποιηθεί με τα δεδομένα, είναι το άτομο με το οποίο σχετίζονται τα δεδομένα. Όπως γίνεται αντιληπτό, αυτό είναι κάτι πολύ πιο υποκειμενικό και πολύπλοκο σε σχέση με το απόρρητο των δεδομένων. Για αυτό η ιδιωτικότητα βασίζεται σε αρχές και καθοδηγείται από κανονισμούς που ορίζονται από τις κυβερνήσεις που εκπροσωπούν τους ανθρώπους.

Το πεδίο εφαρμογής της ιδιωτικότητας περιορίζεται γενικά σε δεδομένα που είναι προσωπικά αναγνωρίσιμα (personally identifiable). Δηλαδή, δεδομένα και τυχόν άλλα δεδομένα περιβάλλοντος με βάση τα

οποία μπορεί να ταυτοποιηθεί ένα άτομο. Τα περισσότερα πρότυπα και κανονισμοί ιδιωτικότητας αναγνωρίζουν ότι εάν τα δεδομένα είναι ανωνυμοποιημένα, σε ορισμένες περιπτώσεις ψευδοανωνυμοποιημένα, τότε δεν εμπίπτουν πλέον στο πεδίο εφαρμογής των απαιτούμενων πρακτικών ή των σχετικών κινδύνων ιδιωτικότητας. Επίσης, οι περισσότεροι κανονισμοί αναγνωρίζουν ένα κεντρικό σύνολο αρχών σύμφωνα με τις οποίες οι χρήστες πρέπει να συναντούν στη συλλογή των πληροφοριών τους, να ενημερώνονται για το πως χρησιμοποιούνται τα δεδομένα τους και να έχουν τη δυνατότητα να υποβάλλουν αιτήματα σχετικά με τη χρήση των δεδομένων τους. Ορισμένοι κανονισμοί (όπως το GDPR [1]) προχωρούν περαιτέρω σε σχέση με τους όρους πώλησης δεδομένων, τη δυνατότητα μεταφοράς δεδομένων χωρίς εμπόδια (φορητότητα δεδομένων) και τον τρόπο με τον οποίο πρέπει να αντιμετωπίζονται τυχόν παραβιάσεις δεδομένων.

Η προστασία της ιδιωτικότητας δεν αναφέρεται μόνο στα ίδια τα δεδομένα, αλλά αφορά και το νόημα που έχουν τα δεδομένα για το άτομο που αφορούν. Εξασφαλίζονται απλά και μόνο έναν καλό έλεγχο του απορρήτου, αυτό δεν διασφαλίζει καθόλου την ιδιωτικότητα. Για παράδειγμα, οι πρώτες απόπειρες προστασίας της ιδιωτικότητας βασίστηκαν στην υπόθεση ότι αρκούσε απλώς η ανωνυμοποίηση των προφανών αναγνωριστικών, αλλά στην πραγματικότητα αποδείχτηκε ότι ακόμη και τα ανώνυμα δεδομένα μπορούν να είναι πολύ χρήσιμα για την ταυτοπόιηση ατόμων [2].

Στα πλαίσια αυτού του κεφαλαίου επικεντρωνόμαστε στην παρουσίαση διάφορων κρυπτογραφικών εφαρμογών που αξιοποιούν διάφορα (προσωπικά) δεδομένα, ενώ χωρίς να τα αποκαλύπτουν, διασφαλίζουν το απόρρητο των δεδομένων ή/και την ιδιωτικότητα των ατόμων. Για αυτό το λόγο, στις ενότητες που ακολουθούν, δεν θα ασχοληθούμε ξανά με την διαφοροποίηση αυτών των εννοιών. Επιπρόσθετα, θα πρέπει να επισημανθεί ότι οι ενότητες που ακολουθούν αφορούν μόνο δημοφιλή και ενδεικτικά μόνο παραδείγματα εφαρμογών διασφάλισης του απορρήτου και της ιδιωτικότητας, ενώ σε καμία περίπτωση αυτές οι εφαρμογές δεν είναι οι μόνες στο χώρο.

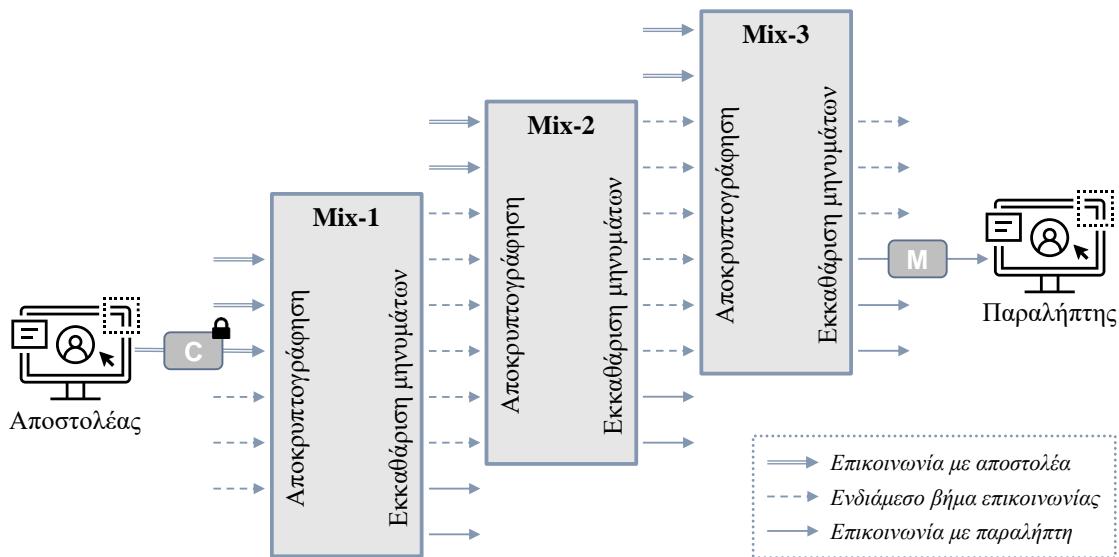
12.2 Δίκτυα Ανωνυμίας

Τα δίκτυα ανωνυμίας (anonymity networks) επιτρέπουν στους χρήστες του Διαδικτύου να διατηρούν ένα επίπεδο ιδιωτικότητας εμποδίζοντας τη συλλογή πληροφοριών αναγνώρισης, όπως οι IP διευθύνσεις. Τα περισσότερα δίκτυα ανωνυμίας αποτελούν απόγονους των δίκτυων μίξης (mix networks) που χρησιμοποιούν μια αλυσίδα διακομιστών μεσολάβησης (proxy) για να δημιουργήσουν δύσκολα ανιχνεύσιμες επικοινωνίες [3]. Οι υπηρεσίες ανωνυμίας παρέχονται είτε από εμπορικές εταιρείες με συνδρομή ή διαφημίσεις, είτε από μη εμπορικές υπηρεσίες μέσω εργαλείων ανωνυμίας ανοιχτού κώδικα. Τέτοια παραδείγματα δίκτυων ανωνυμίας είναι το Java Anon Proxy (JAP) [4], το TOR [5] και το Invisible Internet Project (I2P) [6].

Τα δίκτυα ανωνυμίας στέλνουν τα πακέτα δεδομένων μέσω αναμεταδότων, έτσι ώστε κανένας να μην έχει πληροφορίες τόσο για τον αποστολέα όσο και για τον παραλήπτη των πακέτων. Δεδομένου ότι πολλοί άνθρωποι χρησιμοποιούν ταυτόχρονα τέτοιους είδους μεσάζοντες, η σύνδεση ενός οποιουδήποτε συγκεκριμένου ατόμου στο Διαδίκτυο κρύβεται ανάμεσα στις συνδέσεις όλων των υπόλοιπων χρηστών. Ως εκ τούτου, κανένα μεμονωμένο σύστημα, εσωτερικό ή εξωτερικό του δίκτυου ανωνυμίας, δεν μπορεί να καθορίσει ποια σύνδεση ανήκει σε ποιον χρήστη. Ο βαθμός ανωνυμίας ποικίλλει και εξαρτάται από τους μηχανισμούς που χρησιμοποιούνται, τις δυνατότητες του αντιπάλου και το περιβάλλον λειτουργίας.

12.2.1 Δίκτυα Μίξης (Mix-Nets)

Ο υποκείμενος μηχανισμός των περισσότερων από τα τρέχοντα δίκτυα ανωνυμίας υψηλής καθυστέρησης (high-latency) είναι η μίξη (mix) [3]. Το βασικό δομικό στοιχείο αυτών των συστημάτων, όπως φαίνεται στο Σχήμα 12.1, είναι ένα σύνολο διεργασιών μίξης όπου κάθε διεργασία μίξης λαμβάνει μηνύματα που είναι κρυπτογραφημένα με το δημόσιο κλειδί της εκάστοτε διεργασίας μίξης. Η διεργασία μίξης ομαδοποιεί τα μηνύματα ως μια δέσμη και προωθεί τα κρυπτογραφημένα μηνύματα στην επόμενη διεργασία μίξης σε συγκεκριμένους χρόνους μαζί με περιστασιακά εικονικά μηνύματα.



Σχήμα 12.1: Παράδειγμα επικοινωνίας σε ένα δίκτυο μίξης (Mix-Net).

Τα μηνύματα τελικά φτάνουν στον προορισμό τους αφού προωθηθούν από ένα σύνολο διεργασιών μίξης (βλέπε Σχήμα 12.1) μέσω του δικτύου. Για παράδειγμα, η διαδρομή P ενός μηνύματος M αποτελείται από τρεις διεργασίες μίξης $Mix - 1$, $Mix - 2$ και $Mix - 3$. Ο αποστολέας δημιουργεί ένα κρυπτοκείμενο C που κρυπτογραφεί το μήνυμα M μαζί με έναν τυχαίο κείμενο R χρησιμοποιώντας το δημόσιο κλειδί της κάθε διεργασίας μίξης. Το κρυπτοκείμενο (π.χ. $C = E_1(A_{Mix-2}, R_1 + E_2(A_{Mix-3}, R_2 + E_3(D, R + M)))$) καθορίζει την ακριβή διαδρομή που θα ακολουθήσει το μήνυμα μέσω του δικτύου μίξης. Κάθε κόμβος μιας διεργασίας μίξης (π.χ. ο $Mix - 1$) λαμβάνει το κρυπτοκείμενο το οποίο το αποκρυπτογραφεί κατά ένα επίπεδο για να βρει τον επόμενο προορισμό μετάβασης (π.χ. την διεύθυνση A_{Mix-2}) και του προωθεί το εναπομείναντα κρυπτοκείμενο (π.χ. το $E_2(A_{Mix-3}, R_2 + E_3(D, R + M))$).

Η κρυπτογράφηση δημοσίου κλειδιού και οι αλγόριθμοι εκκαθάρισης (flushing) είναι καθοριστικοί για το παρεχόμενο επίπεδο ανωνυμίας και την απόδοση ενός δικτύου μίξης. Καθώς οι αλγόριθμοι κρυπτογράφησης είναι αποδεδειγμένα ασφαλείς για ένα αρκετά μεγάλο κλειδί, οι αλγόριθμοι εκκαθάρισης είναι ένα σημαντικό στοιχείο που μπορεί να εκθέσει την ταυτότητα των χρηστών. Οι αλγόριθμοι εκκαθάρισης αποθηκεύουν τα εισερχόμενα μηνύματα σε ένα αποθετήριο και προωθούν τα μηνύματα σε γύρους. Σε κάθε γύρο, ένα τυχαίο υποσύνολο των μηνυμάτων του αποθετηρίου αναμειγνύεται με εικονικά μηνύματα και προωθείται. Το υποσύνολο αυτό μπορεί να έχει σταθερό ή δυναμικό πλήθος μηνυμάτων. Η διάρκεια κάθε γύρου αποφασίζεται με βάση ένα όριο. Το όριο αυτό μπορεί να βασίζεται στον αριθμό των μηνυμάτων N στο αποθετήριο, σε έναν μετρητή χρόνου T ή σε συνδυασμό και των δύο. Έχουν αναπτυχθεί διάφορες παραλλαγές των δικτύων μίξης, συμπεριλαμβανομένων του δικτύου Crowds [7], του Tarzan [8] και του JAP [4]. Ωστόσο, σχεδόν κανένα από τα δίκτυα αυτά (με τελευταίο το JAP που τερμάτισε την λειτουργία του το 2021) δεν χρησιμοποιούνται πλέον.

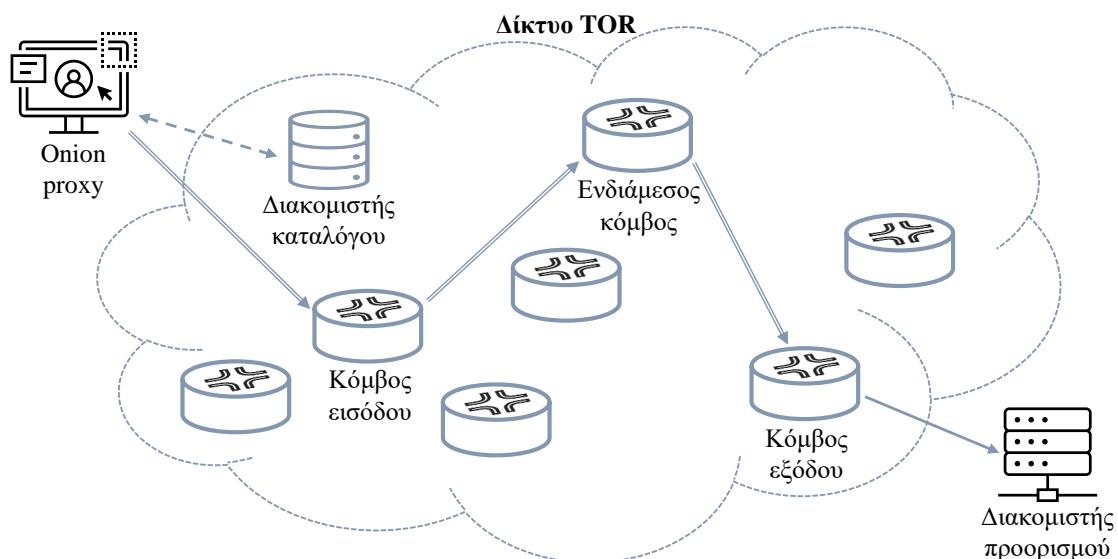
Το JAP (Java Anon Proxy) αποτελεί μια αλληλουχία (cascade) μίξης χαμηλής καθυστέρησης που χρησιμοποιεί διακομιστές που παρέχονται από εθελοντές, συνήθως ιδρύματα που δηλώνουν συμμόρφωση με τις πολιτικές του JAP, για την περιήγηση στο Διαδίκτυο [4]. Οι αλληλουχίες JAP κρυπτογραφούν πακέτα μέσω πολλαπλών μίξεων και διατηρούν την μεταφορά δεδομένων σε σταθερό ρυθμό για να αποφευχθεί η οποιαδήποτε ανάλυση του ρυθμού μεταφοράς. Το πρόγραμμα διαχείρισης του JAP εμφανίζει τις ενεργές μίξεις και οι χρήστες μπορούν να επιλέξουν αλληλουχίες JAP από αυτές τις ενεργές μίξεις. Συνολικά, το JAP έχει πολύ μικρό αριθμό αναμεταδοτών για να παρέχει μια αρκετά ισχυρή ανωνυμία στους χρήστες του. Σημειώνεται ωστόσο, ότι καθώς αυξάνεται ο αριθμός των συμμετεχόντων σε ένα δίκτυο ανωνυμίας, γίνεται όλο και πιο δύσκολο για έναν επιτιθέμενο να παρακολουθήσει τις δραστηριότητες όλων αυτών των χρηστών.

12.2.2 Δρομολόγησης Onion Routing

Η δρομολόγηση Onion Routing αποτελεί μια προσέγγιση ανώνυμης επικοινωνίας χαμηλής καθυστέρησης και αυτή τη στιγμή θεωρείται η πιο διαδεδομένη τεχνική δικτύων ανωνυμίας [9]. Η βασική ιδέα αυτής της δρομολόγησης είναι παρόμοια με αυτή του δικτύου μίξης, αλλά μεταξύ άλλων αλλαγών στον σχεδιασμό, η απόδοσή της βελτιώνεται κυρίως λόγω της χρήσης συμμετρικών κλειδιών για την αναμετάδοση των μηνυμάτων και των ασύμμετρων κλειδιών (κρυπτογραφία δημοσίου κλειδιού) για τη δημιουργία κυκλωμάτων (διαδρομών) στο σύστημα. Υπάρχουν διάφορες παραλλαγές δρομολογητών Onion Routers, όπως το TOR [5] και το I2P [6]. Αυτά τα συστήματα διαφέρουν ανάλογα με τον τρόπο οργάνωσης των διακομιστών δρομολόγησης, το πως εφαρμόζονται οι αλγόριθμοι κρυπτογράφησης, το πως δημιουργούνται τα κρυπτογραφημένα τούνελ (tunnels), εάν το πρωτόκολλο επιτέλουν μεταφοράς χρησιμοποιεί TCP ή UDP πακέτα, ή/και εάν τα ίδια προγράμματα πελάτη (clients) αναμεταδίδουν την κίνηση σε άλλους πελάτες ή όχι.

12.2.2.1 The Onion Router (TOR)

Το TOR [5] αποτελεί ένα δίκτυο ανωνυμίας χαμηλής καθυστέρησης, σχεδιασμένο για εφαρμογές που βασίζονται στο πρωτόκολλο TCP, όπως η περιήγηση στον Διαδίκτυο και η ανταλλαγή άμεσων μηνυμάτων. Συνδύαζει τα καλύτερα στοιχεία των προηγούμενων μεθόδων, όπως την ανεύρεση του καταλόγου διακομιστών δρομολόγησης για τα προγράμματα πελάτη, την δημιουργία δυναμικών κυκλωμάτων και την απόκρυψη τοποθεσίας. Όπως φαίνεται στο Σχήμα 12.2, οι χρήστες του δικτύου TOR εκτελούν ένα πρόγραμμα πελάτη, το οποίο ονομάζεται Onion Proxy, για την σύνδεση με το δίκτυο TOR. Οι συνδέσεις εντός του TOR είναι κρυπτογραφημένες με το πρωτόκολλο TLS, ωστόσο, οι συνδέσεις μεταξύ του TOR κόμβου εξόδου και της διεύθυνσης του προορισμού δεν είναι κρυπτογραφημένες και οι κόμβοι εξόδου ενδέχεται να παρακολουθούν το περιεχόμενο των μηνυμάτων εάν δεν είναι κρυπτογραφημένα από τον χρήστη. Η μεταφορά δεδομένων πραγματοποιείται με πακέτα σταθερού μεγέθους των 512 bytes μέσω του κυκλώματος των TOR κόμβων (οι οποίοι βρίσκονται καταχωρημένοι σε έναν διακομιστή καταλόγου του δικτύου TOR). Κάθε πακέτο περιέχει μια επικεφαλίδα και τα ωφέλιμα δεδομένα. Η επικεφαλίδα αποτελείται από ένα αναγνωριστικό (identifier) κυκλώματος και μια εντολή που περιγράψει τι πρέπει να γίνει με τα ωφέλιμα δεδομένα. Ο κόμβος εισόδου και ο κόμβος εξόδου από το δίκτυο TOR είναι εξαιρετικά σημαντικοί καθώς γνωρίζουν την διεύθυνση προέλευσης και την διεύθυνση προορισμού της επικοινωνίας, αντίστοιχα.



Σχήμα 12.2: Παράδειγμα επικοινωνίας στο δίκτυο TOR.

12.2.2.2 Invisible Internet Project (I2P)

Παρόμοια με το δίκτυο TOR, το I2P [6] δημιουργεί ένα εθελοντικό δίκτυο συστημάτων και προσφέρει υπηρεσίες ανωνυμίας που μπορούν να χρησιμοποιηθούν από ευαίσθητες εφαρμογές σε θέματα ταυτοποίησης. Το δίκτυο I2P βασίζεται αυστηρά σε UDP πακέτα, αλλά υπάρχουν βιβλιοθήκες που παρέχουν πιο αξιόπιστη επικοινωνία πάνω από το δίκτυο I2P. Το I2P λειτουργεί δρομολογώντας την μεταφορά δεδομένων μέσω άλλων ομότιμων κόμβων (peers) και κυρίως είναι ένα κλειστό σύστημα. Πολλές εφαρμογές μπορούν να αλληλεπιδράσουν με το I2P, όπως το email, τα δίκτυα ομότιμων κόμβων (peer-to-peer) και το IRC (Internet Relay Chat). Ωστόσο, καθώς το I2P δεν εστιάζει σε υψηλές ταχύτητες επικοινωνίας, δεν προτιμάται από εφαρμογές που απαιτούν χαμηλές καθυστερήσεις.

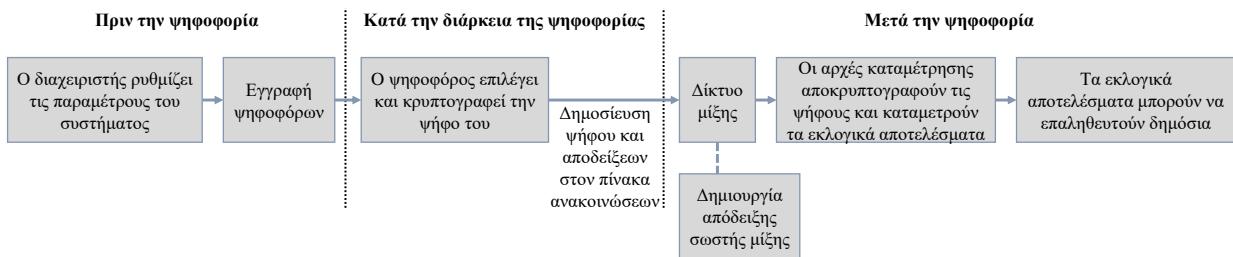
12.3 Ηλεκτρονική Ψηφοφορία

Η ηλεκτρονική ψηφοφορία (e-voting) αποτελεί ένα ηλεκτρονικό σύστημα που επιτρέπει στους χρήστες να λαμβάνουν μια συλλογική απόφαση ή να ψηφίζουν υποψηφίους σε εκλογές. Διαχειρίζεται την εγγραφή ψηφοφόρων, την εισαγωγή ψήφων, την κρυπτογράφηση ψήφων, τη μετάδοση του ψηφοδελτίου στον διακομιστή, την αποθήκευση ψήφων, την καταμέτρηση ψήφων και την κατάταξη των εκλογικών αποτελεσμάτων. Το σύστημα ηλεκτρονικής ψηφοφορίας μπορεί να χρησιμοποιηθεί σε διάφορες εφαρμογές όπως συστήματα ηλεκτρονικής ψηφοφορίας απευθείας καταγραφής (Direct-Recording e-Voting Systems – DRE), συστήματα οπτικής σάρωσης και υπολογιστές δια-συνδεδεμένους στο Διαδίκτυο. Το σύστημα ηλεκτρονικής ψηφοφορίας προσφέρει πιο ακριβή αποτελέσματα εκλογών, ταχύτερη καταγραφή αποτελεσμάτων, ελαχιστοποίηση των ανθρώπινων λαθών, μεγαλύτερη ευελιξία σε άτομα με ειδικές ανάγκες και αυτόματη καταμέτρηση των εκλογικών αποτελεσμάτων [10]. Ωστόσο, σύμφωνα με τους Peng [11], και Oo και Aung [12], η ηλεκτρονική ψηφοφορία αντιμετωπίζει προκλήσεις επεκτασιμότητας για εκλογές μεγάλης κλίμακας, προκλήσεις ασφαλείας, απρόβλεπτες δυσλειτουργίες διακομιστών και άλλα. Επιπλέον, μερικοί άνθρωποι αισθάνονται άβολα να νιοθετήσουν συστήματα ηλεκτρονικής ψηφοφορίας λόγω ζητημάτων σχετικών με την ιδιωτικότητα των ψηφοφόρων, μόνο και μόνο με την υποψία ότι ενδέχεται να αποκαλύφθει η ταυτότητα των ψηφοφόρων. Οι πιο σημαντικές ιδιότητες ασφαλείας που πρέπει να διασφαλίζονται, όπως αναφέρονται από τους Peng [11] και Sebé et al. [13] είναι η ιδιωτικότητα του ψηφοφόρου, η αμεροληψία, η αποδοχή, η αντίσταση στον εξαναγκασμό, η ατομική επαληθευσιμότητα, η καθολική επαληθευσιμότητα, η ανθεκτικότητα, η αποτροπή της διπλής ψηφοφορίας κ.λπ. Έτσι, έχουν προταθεί διάφορα σχήματα για τη βελτίωση της ασφάλειας των συστημάτων ηλεκτρονικής ψηφοφορίας και τα οποία έχουν εφαρμοστεί στην πράξη. Σε αυτήν την ενότητα, εστιάζουμε σε κλασσικές προσεγγίσεις ηλεκτρονικής ψηφοφορίας που βασίζονται σε δίκτυα μίξης (Mix-Nets), στην ομομορφική κρυπτογράφηση, σε τυφλές υπογραφές, καθώς και σε πιο σύγχρονες προσεγγίσεις που βασίζονται σε αλυσίδες μπλοκ (blockchain) και στην μετα-κβαντική κρυπτογραφία.

12.3.1 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Δικτύων Μίξης

Η ηλεκτρονική ψηφοφορία που βασίζεται σε δίκτυα μίξης (βλέπε Ενότητα 12.2.1) προτάθηκε για πρώτη φορά από τον Chaum [3], όπου ο στόχος ενός δικτύου μίξης είναι να δημιουργεί ένα ανώνυμο δικτυακό κανάλι έτσι ώστε η επικοινωνία να παραμένει ανώνυμη. Σε ένα σύστημα ηλεκτρονικής ψηφοφορίας, το δίκτυο μίξης αποτέλεπει την αντιστοίχηση μεταξύ των ψηφοφόρων και των ψηφοδελτίων τους. Όπως τονίζεται από τον Jakobsson et al. [14], το δίκτυο μίξης θα πρέπει να παρέχει ισχυρή ανωνυμία, ώστε να εγγυάται την ιδιωτικότητα και την σωστή λειτουργία του. Στο Σχήμα 12.3 αποτυπώνεται η γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας που βασίζεται σε δίκτυα μίξης πριν από την ψηφοφορία, κατά την διάρκεια της ψηφοφορίας και μετά την ψηφοφορία. Οι διαδικασίες στη φάση της προεκλογικής διαδικασίας περιλαμβάνουν την ανάδειξη υποψηφίων, τον υπολογισμό του καταλόγου των υποψηφίων, την εγγραφή ψηφοφόρων και τον υπολογισμό του καταλόγου των επιλέξιμων ψηφοφόρων. Οι ψηφοφόροι που έχουν δικαιώμα ψήφου ψηφίζουν κατά την φάση της ψηφοφορίας. Η φάση μετά την ψηφοφορία αφορά κυρίως την καταμέτρηση των

ψήφων και την ανακοίνωση των εκλογικών αποτελεσμάτων.



Σχήμα 12.3: Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση δικτύων μίξης.

Τα σχήματα ηλεκτρονικής ψηφοφορίας με χρήση δικτύων μίξης μπορούν να χωριστούν σε δύο κατηγορίες [15], στα δίκτυα μίξης με αποκρυπτογράφηση και στα δίκτυα μίξης με επανα-κρυπτογράφηση (re-encryption). Το πρώτο σύστημα ηλεκτρονικής ψηφοφορίας που προτάθηκε από τον Chaum [3] ανήκε στην πρώτη κατηγορία σχημάτων και χρησιμοποιούσε RSA αποκρυπτογράφηση [16]. Σε αυτό, κάθε διακομιστής μίξης κατέχει ένα ζεύγος RSA κλειδιών, και ο αποστολέας κρυπτογραφεί το ψηφοδέλτιο του επαναληπτικά με τα δημόσια κλειδιά των διακομιστών μίξης με αντίστροφη σειρά από αυτή της μετάδοσης του (γνωστή ως κρυπτογράφηση onion). Ο πρώτος διακομιστής μίξης αποκρυπτογραφεί το εξωτερικό επίπεδο κρυπτογράφησης των κρυπτοκειμένων, τα ανακατεύει και τα μεταβιβάζει στον επόμενο διακομιστή. Ο δεύτερος διακομιστής μίξης επαναλαμβάνει την ίδια διαδικασία με τον πρώτο διακομιστή μίξης και η διαδικασία ολοκληρώνεται όταν όλοι οι διακομιστές μίξης εκτελέσουν αυτήν τη διαδικασία. Στο τέλος, έχουν αφαιρεθεί όλες οι κρυπτογραφήσεις και τα ψηφοδέλτια δημοσιεύονται με τυχαία σειρά.

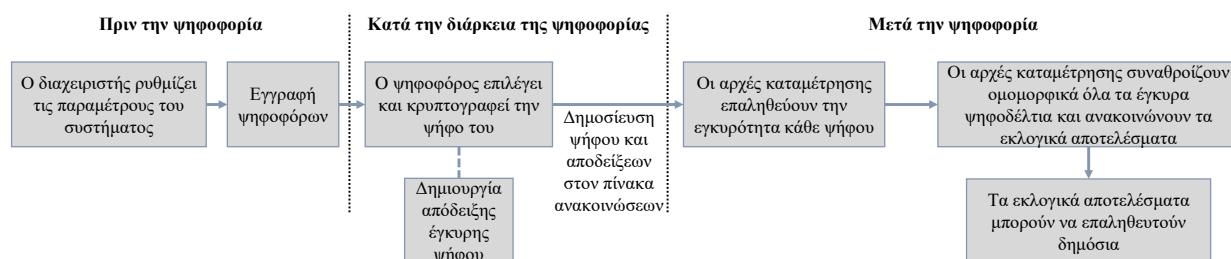
Η δεύτερη κατηγορία δικτύων μίξης με επανα-κρυπτογράφηση προτάθηκε για πρώτη φορά από τον Park et al. [17] και βασίζεται στην τυχαιοποίηση. Αυτά διαθέτουν δύο κύριες φάσεις: την φάση μίξης και την φάση αποκρυπτογράφησης. Στη φάση μίξης, τα κρυπτογραφημένα ψηφοδέλτια ανακατεύονται και επανακρυπτογραφούνται. Στη φάση αποκρυπτογράφησης, αποκρυπτογραφείται απλά η έξοδος από τη φάση μίξης. Ο διακομιστής στις φάσεις μίξης και αποκρυπτογράφησης μπορεί να είναι είτε ο ίδιος είτε διαφορετικός. Το σχήμα που προτάθηκε από τον Park et al. [17] χρησιμοποιεί το κρυπτοσύστημα ElGamal [18] και λειτουργεί με την διαδικασία που περιγράφεται παρακάτω. Σε αυτό, υπάρχουν πολλοί διαχειριστές της ψηφοφορίας που διαμοριάζονται το ιδιωτικό κλειδί (δηλ., ο καθένας του κατέχει ένα μέρος του κλειδιού και κατ' επέκταση απαιτούνται όλοι στο τέλος για να αποκρυπτογραφήσουν το αποτέλεσμα) και ο αποστολέας κρυπτογραφεί το ψηφοδέλτιο του με το κοινό δημόσιο κλειδί. Ο πρώτος διακομιστής μίξης επανα-κρυπτογραφεί το κρυπτοκειμένο του αποστολέα, το ανακατεύει και το στέλνει στους επόμενους διακομιστές μίξης. Όλοι οι διακομιστές μίξης επαναλαμβάνουν την ίδια διαδικασία μια φορά και τα αποτελέσματα δημοσιεύονται με τυχαία σειρά μετά την αποκρυπτογράφησή τους από τους διαχειριστές.

Τα πιο πρόσφατα συστήματα ηλεκτρονικής ψηφοφορίας [15] που έχουν προταθεί χρησιμοποιούν ως επί το πλείστον δίκτυα μίξης με επανα-κρυπτογράφηση καθώς είναι πιο αποτελεσματικά, ανθεκτικά και ευέλικτα. Επιπλέον, τα δίκτυα μίξης με επανα-κρυπτογράφηση είναι πιο ελαφριά από τα δίκτυα μίξης με αποκρυπτογράφηση, καθώς το ψηφοδέλτιο κρυπτογραφείται μόνο μια φορά με το δημόσιο κλειδί, ενώ στα δίκτυα μίξης με αποκρυπτογράφηση το ψηφοδέλτιο κρυπτογραφείται επαναληπτικά σε επίπεδα (κρυπτογράφηση onion). Σε ένα δίκτυο μίξης με επανα-κρυπτογράφηση, μια μεμονωμένη λάθος μίξη δεν επηρεάζει τη διαδικασία εκλογής, σε αντίθεση με ένα δίκτυο μίξης με αποκρυπτογράφηση. Ωστόσο, στο δίκτυο μίξης με αποκρυπτογράφηση, απαιτείται να επιλεγεί ένα σταθερό σύνολο μίξεων (και επιπλέον να είναι διαθέσιμα τα κλειδιά τους πριν από την ψηφοφορία), γεγονός που οδηγεί στην αποτυχία των δικτύων μίξης με αποκρυπτογράφηση στο να ολοκληρώσουν την εκλογική διαδικασία εάν υπάρχει έστω και μια λάθος μίξη.

12.3.2 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Ομομορφικής Κρυπτογράφησης

Η ομομορφική κρυπτογράφηση (βλέπε Ενότητα 8.2) επιτρέπει την πραγματοποίηση πράξεων στα κρυπτοκείμενα χωρίς να απαιτείται η αποκρυπτογράφησή τους. Για παράδειγμα, εάν υποθέσουμε ότι υπάρχουν δύο κρυπτοκείμενα $E_K(m_1)$ και $E_K(m_2)$, τότε μπορεί να γίνει η πράξη $E_K(m_1 \odot m_2)$, όπου το \odot μπορεί να είναι είτε η πράξη της πρόσθεσης \oplus , είτε η πράξη του πολλαπλασιασμού \otimes . Υπάρχουν δύο τύποι ομομορφικών σχημάτων: η μερική ομομορφική κρυπτογράφηση και η πλήρης ομομορφική κρυπτογράφηση. Ένα σχήμα μερικής ομομορφικής κρυπτογράφησης μπορεί να εκτελεί μόνο μια πράξη σε επίπεδο κρυπτοκειμένων. Το κρυπτοσύστημα Paillier [19] και το κρυπτοσύστημα ElGamal [18] αποτελούν δύο αρκετά γνωστά σχήματα μερικής ομομορφικής κρυπτογράφησης. Ωστόσο, η δημιουργία κατανεμημένων κλειδιών για το κρυπτοσύστημα El-Gamal είναι πιο αποτελεσματική από το κρυπτοσύστημα Paillier όταν απαιτείται το ίδιο επίπεδο ασφάλειας [20]. Ο ElGamal χρησιμοποιείται συχνότερα σε σχήματα ηλεκτρονικής ψηφοφορίας με χρήση ομομορφικής κρυπτογράφησης, ενώ ένα πλήρως ομομορφικό σχήμα προτάθηκε για πρώτη φορά το 2009 από τον Gentry [21]. Ένα πλήρως ομομορφικό σχήμα κρυπτογράφησης μπορεί να πραγματοποιεί ταυτόχρονα πράξεις πρόσθεσης και πολλαπλασιασμού στα κρυπτοκειμένα.

Η ηλεκτρονική ψηφοφορία με χρήση ομομορφικής κρυπτογράφησης αποτελείται κυρίως από δύο παραλαγές [15], η μια παραλλαγή κάνει χρήση της προσθετικής ομομορφικής ιδιότητας \oplus που προτάθηκε για πρώτη φορά από τους Cohen και Fischer [22] και η δεύτερη της πολλαπλασιαστικής ομομορφικής ιδιότητας \otimes που προτάθηκε για πρώτη φορά από τους Peng *et al.* [23]. Η ηλεκτρονική ψηφοφορία με χρήση ομομορφικής κρυπτογράφησης είναι κατάλληλη για εκλογές μικρής κλίμακας (π.χ. για εκλογές ΝΑΙ/ΟΧΙ). Η διαφορά μεταξύ της προσθετικής και πολλαπλασιαστικής παραλλαγής της ηλεκτρονικής ψηφοφορίας βρίσκεται μόνο κατά την φάση της καταμέτρησης των ψήφων. Στη φάση της καταμέτρησης με την προσθετική ομομορφική ιδιότητα, το άθροισμα των ψήφων ανακτάται για τους υποψηφίους ως $E_K(m_1) \cdots E(m_n) = E_K(m_1 + \cdots + m_n)$ και σε καμία περίπτωση δεν αποκρυπτογραφούνται οι ίδιοι οι ψήφοι, παρά μόνο το τελικό αποτέλεσμα. Στη φάση της καταμέτρησης με την πολλαπλασιαστική ομομορφική ιδιότητα, αποκρυπτογραφείται το γινόμενο των ψήφων $E_K(m_1) \cdots E(m_n) = E_K(m_1 \cdots m_n)$ και στη συνέχεια υπολογίζεται η ρίζα του γινομένου $\sqrt{m_1 \cdots m_n}$, όπου z ο αριθμός που χρησιμοποιήθηκε για να αντιπροσωπεύσει μια θετική ψήφο, έτσι ώστε να βρεθεί το πλήθος των ψήφων. Στο Σχήμα 12.4 αποτυπώνεται η γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας που βασίζεται στην ομομορφική κρυπτογράφηση πριν από την ψηφοφορία, κατά την διάρκεια της ψηφοφορίας και μετά την ψηφοφορία.

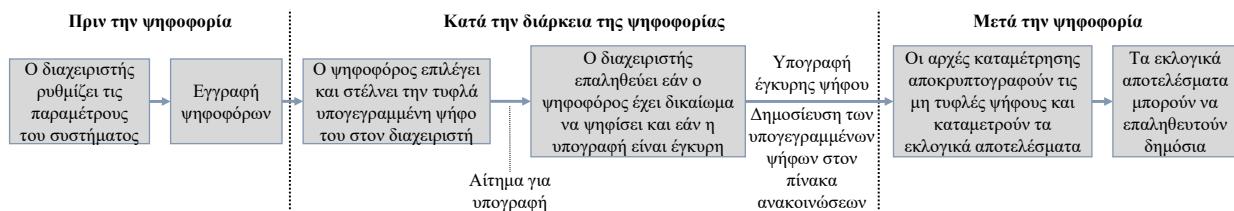


Σχήμα 12.4: Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση ομομορφικής κρυπτογράφησης.

12.3.3 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Τυφλών Υπογραφών

Μια τυφλή υπογραφή αποτελεί μια ειδική κατηγορία ψηφιακών υπογραφών ενίσχυσης του απορρήτου (βλέπε Ενότητα 8.4), κατά την οποία το περιεχόμενο ενός μηνύματος συγκαλύπτεται (*blinded*) πριν υπογραφεί. Προτάθηκαν για πρώτη φορά από τον Chaum [24] για ένα μη-ανιχνεύσιμο (*untraceable*) σύστημα πληρωμών. Οι Fujioka *et al.* [25] ήταν αυτοί που για πρώτη φορά εφάρμοσαν ένα σχήμα τυφλών υπογραφών σε ένα σύστημα ηλεκτρονικής ψηφοφορίας. Στο Σχήμα 12.5 παρουσιάζεται η γενική δομή των φάσεων μιας ηλε-

κτρονικής ψηφοφορίας που βασίζεται σε τυφλές υπογραφές πριν από την ψηφοφορία, κατά την διάρκεια της ψηφοφορίας και μετά την ψηφοφορία. Η ηλεκτρονική ψηφοφορία με χρήση τυφλών υπογραφών επιτρέπει στον ψηφοφόρο να συγκαλύψει (blind) την ψήφο του, και με αυτόν τον τρόπο η εκλογική αρχή μπορεί να επικυρώσει την ψήφο χωρίς να γνωρίζει το περιεχόμενό της. Υπάρχουν διάφοροι τύποι τυφλών υπογραφών, όπως οι τυφλές υπογραφές κατωφλίου (threshold) και τυφλές υπογραφές βάσει ταυτότητας (identity-based) [15]. Το σχήμα τυφλής υπογραφής κατωφλίου αποφεύγει την αστοχία ενός μόνο σημείου και έτσι ενισχύει την ανθεκτικότητα του σχήματος, ενώ η διαδικασία υπογραφής πραγματοποιείται συνολικά N φορές μεταξύ των οντότητων. Επιπλέον, προϋποθέτει ότι τουλάχιστον t υπογραφές πραγματοποιήθηκαν σωστά, όπου το όριο t πρέπει να είναι μεγαλύτερο από 1 και μικρότερο από N . Η διαδικασία υπογραφής στο συγκαλυμμένο (blind) μήνυμα πραγματοποιείται από καθεμία από τις N οντότητες και μόνο εάν το μήνυμα υπογραφεί από t οντότητες, θεωρείται έγκυρη η υπογραφή. Σχήμα τυφλής υπογραφής βάσει ταυτότητας προτάθηκε για πρώτη φορά από τους Zhang και Kim [26], και οι Kumar *et al.* [27] εφάρμοσαν κατόπιν ένα τέτοιο σχήμα σε ένα σύστημα ηλεκτρονικής ψηφοφορίας. Σε αυτό στο σχήμα, το σύστημα εκδίδει μια απόδειξη στον ψηφοφόρο και αυτές οι απόδειξεις ψηφοφορίας μπορούν να χρησιμεύσουν με την σειρά τους ως απόδειξη συμμετοχής. Ωστόσο, κάτι τέτοιο θα μπορούσε να αποτελέσει πρόβλημα, γιατί θα μπορούσαν οι απόδειξεις συμμετοχής να μεταβιβαστούν σε άλλα άτομα που δεν πήραν μέρος στην διαδικασία ψηφοφορίας, μιας και δεν παρέχουν σύνδεση με μια φυσική οντότητα.



Σχήμα 12.5: Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση τυφλών υπογραφών.

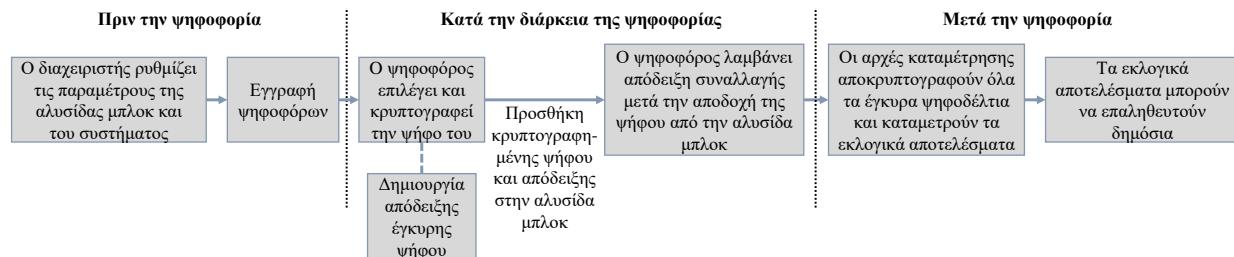
Τα συστήματα ηλεκτρονικής ψηφοφορίας που χρησιμοποιούν τυφλές υπογραφές ως βασικό εργαλείο ενίσχυσης του απορρήτου πάσχουν από το πρόβλημα της αποχής των ψηφοφόρων, και επιπλέον, είναι δύσκολο να σχεδιαστεί ένα σύστημα τυφλών υπογραφών που να μην επιτρέπει σε μια διεφθαρμένη εκλογική αρχή να προσθέτει ψήφους της επιλογής της [15]. Αυτό το πρόβλημα, ωστόσο, θα μπορούσε να επιλυθεί με τη χρήση πολλαπλών εκλογικών αρχών, έτσι ώστε μια διεφθαρμένη αρχή να μην έχει την εξουσία να ελέγχει ολόκληρη την εκλογική διαδικασία.

12.3.4 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Αλυσίδων Μπλοκ

Στην ηλεκτρονική ψηφοφορία με την χρήση αλυσίδων μπλοκ (βλέπε Κεφάλαιο 11), η αλυσίδα μπλοκ αποθηκεύει τα ψηφοδέλτια [28] και οι ψήφοι που αποθηκεύονται σε αυτήν δεν μπορούν να διαγραφούν ή να τροποποιηθούν. Και αυτό γιατί η αλυσίδα μπλοκ είναι αμετάβλητη (immutable), καθώς το κάθε μπλοκ αποτελείται από την σύνοψη του προηγούμενου μπλοκ, και επομένως όλα τα μπλοκ συνδέονται μεταξύ τους. Ωστόσο, η ηλεκτρονική ψηφοφορία που βασίζεται σε αλυσίδες μπλοκ είναι ακόμη σε πρώιμο στάδιο και δεν έχει εφαρμοστεί πλήρως σε εκλογές μεγάλης κλίμακας [29]. Στο Σχήμα 12.6 παρουσιάζεται η γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας που βασίζεται σε αλυσίδες μπλοκ πριν από την ψηφοφορία, κατά την διάρκεια της ψηφοφορίας και μετά την ψηφοφορία.

Οι Liu και Wang [30] αναφέρουν ότι ορισμένα από τα σχήματα ηλεκτρονικής ψηφοφορίας που έχουν προταθεί περιλαμβάνουν την ύπαρξη μιας αξιόπιστης τρίτης οντότητας επειδή είναι σχετικά απλό να ελεγχθεί και να εφαρμοστεί στο σύστημα, αλλά είναι επίσης πιθανόν μια τέτοια οντότητα να αποτελέσει ταυτόχρονα το εναίσθητο σημείο του συστήματος. Ωστόσο, στην ηλεκτρονική ψηφοφορία με την χρήση αλυσίδων μπλοκ μπορεί να επιτευχθεί η υλοποίηση του συστήματος χωρίς την ύπαρξη κάποιας αξιόπιστης τρίτης οντότητας, το οποίο θα μπορεί να εγγυάται την επαληθευσιμότητα αλλά και την ανωνυμία. Επιπλέον, η τεχνολογία αλυ-

σίδων μπλοκ παρέχει διαφάνεια, επομένως ολόκληρη η εκλογική διαδικασία μπορεί να είναι διαφανής στο κοινό, και αυτό προσφέρει εγκυρότητα και δικαιοσύνη.



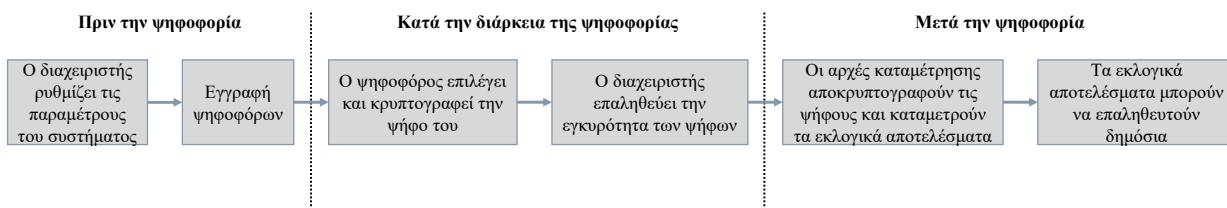
Σχήμα 12.6: Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση αλυσίδων μπλοκ.

Από την άλλη πλευρά, η ηλεκτρονική ψηφοφορία που βασίζεται σε αλυσίδες μπλοκ εισάγει πρόσθετα προβλήματα στα συστήματα ηλεκτρονικής ψηφοφορίας [31, 32]. Παρά τα θεμελιώδη ζητήματα ασφαλείας των εκλογών που μπορούν να επιλυθούν με την εισαγωγή της τεχνολογίας αλυσίδων μπλοκ στα συστήματα ψηφοφορίας, αυτό επιβάλλει επίσης νέες δυσκολίες στο σύστημα. Η αποκεντρωμένη φύση των αλυσίδων μπλοκ αυξάνει σημαντικά την πολυπλοκότητα του συστήματος [31]. Αυτό οδηγεί σε δυσκολίες στη διαχείριση του συστήματος και απαιτείται περισσότερος χρόνος για την επίλυση ή την εφαρμογή των επιδιορθώσεων ασφαλείας σε ένα αποκεντρωμένο σύστημα. Τα ψηφοδέλτια που είναι αποθηκευμένα σε μια αλυσίδα μπλοκ είναι δύσκολο να επαληθευτούν και απαιτούν λογισμικό για τη διαδικασία επαλήθευσης. Η επαληθευσιμότητα μπορεί επίσης να εξαπατηθεί εάν το λογισμικό έχει παραβιαστεί. Επομένως, η αποφυγή ενός τέτοιου λογισμικού είναι δύσκολο να επιτευχθεί εάν τα ψηφοδέλτια αποθηκεύονται σε αλυσίδες μπλοκ [32].

Επιπρόσθετα, το αμετάβλητο των αλυσίδων μπλοκ αποτελεί μια σημαντική πρόκληση για την ακεραιότητα των συστημάτων ψηφοφορίας. Στο σενάριο όπου πραγματοποιείται μια αλλαγή της ψήφου ενός ψηφοφόρου πριν αυτή φτάσει στην αλυσίδα μπλοκ, ο ψηφοφόρος δεν μπορεί να γνωρίζει αυτήν την αλλαγή, η οποία θα προκαλέσει εσφαλμένο αποτέλεσμα στο εκλογικό σύστημα [32]. Επιπλέον, μια αλυσίδα μπλοκ με άδεια (permissioned) δεν μπορεί να ικανοποιήσει τις απαιτήσεις επαληθευσιμότητας της ηλεκτρονικής ψηφοφορίας. Και αυτό γιατί οι ψηφοφόροι δεν μπορούν να διαβάσουν ή να επαληθεύσουν εάν οι ψήφοι τους συμπεριλήφθηκαν στην τελική καταμέτρηση [31].

12.3.5 Συστήματα Ηλεκτρονικής Ψηφοφορίας με Χρήση Μετα-Κβαντικής Κρυπτογραφίας

Η αξιοποίηση της μετα-κβαντικής κρυπτογραφίας (βλέπε Κεφάλαιο 15) σε σχήματα ηλεκτρονικής ψηφοφορίας είναι μια σχετικά νέα ερευνητική κατεύθυνση και λίγοι την έχουν εφαρμόσει στην ηλεκτρονική ψηφοφορία. Η πλήρης ομομορφική κρυπτογράφηση (π.χ. [33]) και η κρυπτογραφία που βασίζεται σε πλέγματα (lattice-based) (π.χ. [34]) είναι κάποιες από τις πιο κοινές τεχνικές που χρησιμοποιούνται για την κατασκευή ενός συστήματος ηλεκτρονικής ψηφοφορίας με την χρήση μετα-κβαντικής κρυπτογραφίας. Η μετα-κβαντική κρυπτογραφία βασίζεται σε διάφορες παραδοχές ανθεκτικότητας που είναι συμβατές με τους κβαντικούς υπολογιστές, όπως οι πολυμεταβλητές γραμμικές εξισώσεις και τα πλέγματα [35]. Από την άλλη, η ασφάλεια των σχημάτων που βασίζονται στην υπολογιστική πολυπλοκότητα ή σε άλλες κλασικές παραδοχές ασφαλείας δεν είναι ασφαλείς αναφορικά με κβαντικές επιθέσεις που πλέον μπορούν να πραγματοποιηθούν λόγω της προόδου των κβαντικών υπολογιστών [36]. Στο Σχήμα 12.7 αποτυπώνεται η γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας που βασίζεται στην μετα-κβαντική κρυπτογραφία πριν από την ψηφοφορία, κατά την διάρκεια της ψηφοφορίας και μετά την ψηφοφορία.



Σχήμα 12.7: Γενική δομή των φάσεων μιας ηλεκτρονικής ψηφοφορίας με χρήση μετα-κβαντικής κρυπτογραφίας.

12.4 Δημοπρασίες με Διασφάλιση Ιδιωτικότητας

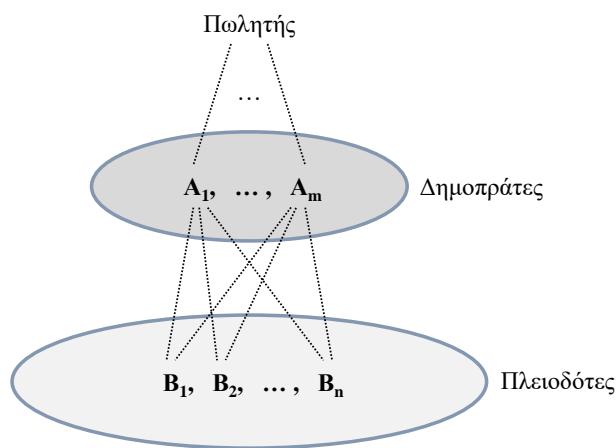
Μια δημοπρασία αποτελεί έναν μηχανισμό διαπραγμάτευσης (συνήθως της τιμής) εμπορευμάτων μεταξύ δύο ομάδων, των πωλητών και των πλειοδοτών (bidders). Οι περισσότερες δημοπρασίες περιλαμβάνουν επίσης έναν δημοπράτη (auctioneer) που είναι υπεύθυνος για τη διευθέτηση της δημοπρασίας, την αποδοχή των προσφορών και την ανακήρυξη νικητή εκ μέρους του πωλητή. Για να εκτελεστεί σωστά μια δημοπρασία, πρέπει να υπάρχουν μέθοδοι εγγραφής συμμετεχόντων, αποδοχής προσφορών και ανοίγματος προσφορών. Η μέθοδος που χρησιμοποιείται για την υποβολή προσφορών καθορίζει τον τύπο της δημοπρασίας [37]. Για παράδειγμα, εάν μια δημοπρασία απαιτεί από τους συμμετέχοντες να υποβάλλουν προσφορές με αυξανόμενο τρόπο, αυτή λέγεται ότι εμπίπτει στην κατηγορία της αγγλικής δημοπρασίας (English auction). Από την άλλη πλευρά, εάν ακολουθηθεί η αντίθετη προσέγγιση, ο μηχανισμός χαρακτηρίζεται ως ολλανδική δημοπρασία (Dutch auction), δηλαδή, η τιμή μειώνεται επανειλημμένα έως ότου κάποιος είναι διατεθειμένος να πληρώσει την τρέχουσα τιμή (αυτό παρατηρείται συνήθως σε ευπαθείς αγορές). Σε έναν άλλο τύπο δημοπρασίας, οι αγοραστές κάνουν προσφορές για ένα υποσύνολο αντικειμένων, κάτι το οποίο είναι γνωστό ως συνδυαστική δημοπρασία (combinatorial auction). Σε γενικές γραμμές, σε έναν μηχανισμό δημοπρασίας, ο νικητής είναι ο πλειοδότης που έχει υποβάλει την υψηλότερη προσφορά. Για τον καθορισμό της τιμής πώλησης, υπάρχουν δύο κυρίως μέθοδοι: η δημοπρασία πρώτης τιμής (first-price) και η δημοπρασία δεύτερης τιμής (second-price). Στην πρώτη, ο νικητής πληρώνει το ποσό που έχει προτείνει, δηλαδή την υψηλότερη προσφορά, ενώ στην δεύτερη, ο νικητής πληρώνει το ποσό της δεύτερης υψηλότερης προσφοράς.

Στα πρωτόκολλα ψηφιακών δημοπρασιών που διασφαλίζουν την ιδιωτικότητα, γνωστά και ως δημοπρασίες σφραγισμένης προσφοράς (sealed-bid), οι πλειοδότες σφραγίζουν τις προσφορές τους χρησιμοποιώντας κάποια κρυπτογραφική τεχνική. Μετά την εκτέλεση της δημοπρασίας, αποκαλύπτονται μόνο τα αποτελέσματα της δημοπρασίας, δηλαδή ο νικητής και η τιμή πώλησης. Αυτό έχει ως αποτέλεσμα, οι υπόλοιπες προσφορές που δεν κέρδισαν να παραμένουν κρυφές. Το κύριο κίνητρο για την κατασκευή πρωτοκόλλων δημοπρασίας σφραγισμένης προσφοράς είναι το γεγονός ότι οι δημοπρασίες ή/και οι πωλητές να μπορούν να χρησιμοποιήσουν τις προηγούμενες χαμένες προσφορές για να μεγιστοποιήσουν τα έσοδά τους σε μελλοντικές δημοπρασίες και διαπραγματεύσεις ανάλογων αντικειμένων. Επιπλέον, οι ιδιωτικές προσφορές των πλειοδοτών μπορούν να χρησιμοποιηθούν για την αποκάλυψη προσωπικών προτιμήσεων και προσωπικών πληροφοριών σχετικά με αυτούς. Επιπρόσθετα, για παράδειγμα, ένα υποσύνολο των προσφορών με τις υψηλότερες χαμένες προσφορές ή ο μέσος όρος αυτών μπορεί να παρακινήσει τους πωλητές ή τους δημοπράτες να αυξήσουν την τιμή εκκίνησης ή την ελάχιστη αξία της προσφοράς σε μελλοντικές δημοπρασίες παρόμοιων αντικειμένων [37].

12.4.1 Δημοπρασίες Πρώτης Τιμής με Σφραγισμένη Προσφορά

Σε μια δημοπρασία πρώτης τιμής με σφραγισμένη προσφορά (First-Price Sealed-Bid Auctions – FPSBA), οι συμμετέχοντες υποβάλλουν ταυτόχρονα τις προσφορές τους σε σφραγισμένη μορφή. Κανένας πλειοδότης δεν θα μάθει τίποτα για το περιεχόμενο άλλης προσφοράς εκτός από τη δική του προσφορά. Μερικά από τα πρώτα κρυπτογραφικά πρωτόκολλα σχετικά με το θέμα αυτό έχουν δημοσιευθεί στα [38, 39, 40, 41, 42].

Για παράδειγμα, οι Kikuchi *et al.* [38] πρότειναν ένα πρωτόκολλο που βασίζεται σε έναν ασφαλή υπολογισμό πολλαπλών οντοτήτων (Secure Multi-Party Computation – MPC) για την εκτέλεση αυτού του τύπου δημοπρασίας. Αυτό το πρωτόκολλο ακολουθεί ένα μοντέλο παρόμοιο με αυτό που πρότειναν οι Franklin και Reiter [43], δηλαδή υπάρχει ένας πωλητής, πολλοί πλειοδότες (B_j) και πολλοί δημοπράτες (A_i , όπου $|A| < |B|$). Στο Σχήμα 12.8 αποτυπώνεται η γενική αρχιτεκτονική αυτού του τύπου δημοπρασιών. Σε αυτό το πρωτόκολλο, ένα σύνολο τιμών k δημοσιεύεται κατά τη φάση προετοιμασίας, και οι πλειοδότες μπορούν να έχουν την επιλογή να εκχωρήσουν ένα αναγνωριστικό (ID) ή μηδέν σε κάθε τιμή k ανάλογα με τις εκτιμήσεις τους για το αντικείμενο. Μόλις ένας πλειοδότης προετοιμάσει μια ακολουθία προσφορών για κάθε k , στην συνέχεια αυτή η ακολουθία γίνεται είσοδος σε έναν ασφαλή υπολογισμό πολλαπλών οντοτήτων (MPC), και το πρωτόκολλο MPC θα έχει ως έξοδο τον νικητή. Για μια τιμή k , εάν υπάρχει μόνο ένας νικητής, τότε το MPC αποκαλύπτει την ταυτότητά του, διαφορετικά, αποκαλύπτει το άθροισμα των ταυτοτήτων τους (το οποίο δεν είναι αναγνωρίσιμο). Όταν δεν βρεθεί νικητής σε μια δεδομένη τιμή k , το αποτέλεσμα είναι μηδέν. Εάν προκύψει ισοπαλία, το οποίο είναι πιθανό για μεγάλα εύρη τιμών k , οι επόμενοι γύροι του πρωτοκόλλου κατασκευάζονται για το εύρος τιμών στο οποίο βρέθηκαν οι νικητές από τον προηγούμενο γύρο.



Σχήμα 12.8: Γενική αρχιτεκτονική μιας δημοπρασίας πρώτης τιμής με σφραγισμένη προσφορά.

Ένα άλλο γενικό σχήμα που ισχύει για τη δημοπρασία πρώτης τιμής, προτάθηκε από τον Cachin [39]. Η ιδιωτικότητα μιας προσφοράς επιτυγχάνεται αυτήν την φορά με συνδυασμό της ομομορφικής κρυπτογράφησης και ενός πρωτοκόλλου MPC. Το πρωτόκολλο δεν εξαρτάται από την αξιολόγηση ενός MPC λογικού κυκλώματος (βλέπε Ενότητα 8.1), ωστόσο, απαιτεί τη χρήση δύο διακομιστών με τους οποίους οι χρήστες επικοινωνούν μόνο με έναν από αυτούς. Αρχικά, η ομομορφική κρυπτογράφηση δημόσιου κλειδιού χρησιμοποιείται για τη σφράγιση των προσφορών. Μέσω ενός πρωτοκόλλου MPC, μια ημι-έμπιστη τρίτη οντότητα συγκρίνει τις προσφορές δύο διαφορετικών πλειοδοτών και, στη συνέχεια, προσδιορίζει την υψηλότερη από τις δύο τιμές κάνοντας σύγκριση τα κρυπτογραφημένα bits με σειρά προτεραιότητας από τα πιο σημαντικά ψηφία προς τα λιγότερο σημαντικά. Στο τέλος του πρωτοκόλλου, αποκαλύπτεται μόνο η υψηλότερη προσφορά, ωστόσο το πρωτόκολλο μπορεί εύκολα να επεκταθεί για να βρεθεί και η δεύτερη υψηλότερη προσφορά. Ένα από τα μειονεκτήματα αυτού του πρωτοκόλλου είναι ότι η σύγκριση πρέπει να πραγματοποιηθεί ανά ξενή τιμών σε κάθε βήμα.

Σε μια άλλη προσέγγιση [44], προτείνεται μια διαφορετική λύση κάνοντας χρήση δεσμευτικών ομαδικών υπογραφών (group signatures) (βλέπε Ενότητα 8.4). Σε αυτό το πρωτόκολλο, κάθε πλειοδότης πρέπει να λάβει ένα πιστοποιητικό που του επιτρέπει να συμμετάσχει στη δημοπρασία. Στη συνέχεια, χρησιμοποιώντας αυτό το ιδιωτικό πιστοποιητικό, ένας συγκεκριμένος πλειοδότης μπορεί να υποβάλει μια προσφορά και να την υπογράψει με ένα σχήμα ομαδικής υπογραφής. Αυτή η ομαδική υπογραφή της προσφοράς διανέμεται χρησιμοποιώντας ένα σχήμα κοινής χρήσης ομαδικών υπογραφών. Κατά τη φάση του ανοίγματος, οι δημοπράτες ανακτούν την υψηλότερη προσφορά χρησιμοποιώντας έναν MPC υπολογισμό. Στο τέλος του πρωτοκόλλου,

μπορεί να βρεθεί η ταυτότητα που σχετίζεται με τη νικήτρια προσφορά μέσω μιας διαδικασίας ανάκλησης του συστήματος ομαδικών υπογραφών.

Τέλος, μια άλλη ενδιαφέρουσα λύση [45] περιγράφει ένα σχήμα δέσμευσης πολλαπλών στοιχείων που αποτελείται από μια αξιόπιστη οντότητα και n προσφορές. Κατά τη φάση προετοιμασίας, η αξιόπιστη οντότητα επιλέγει n πολυώνυμα βαθμού $n - 1$, στέλνει ένα πολυώνυμο $g_i(x)$ στον πλειοδότη B_i , και επίσης στέλνει $n - 1$ διακριτά σημεία για κάθε $g_i(x)$ στους υπόλοιπους πλειοδότες. Κάθε πλειοδότης B_i υπολογίζει το $y_i = g_i(\beta_i)$ ως μια δεσμευμένη τιμή για την προσφορά του β_i και μεταδίδει το y_i στους υπόλοιπους πλειοδότες. Για τη διεξαγωγή αυτού του σχήματος δημοπρασίας χρησιμοποιείται ο τύπος της ολλανδικής δημοπρασίας (Dutch auction). Κατά τη φάση της αποκάλυψης, ένας πλειοδότης ισχυρίζεται ότι είναι ο νικητής σε μια συγκεκριμένη τιμή και στη συνέχεια αποδεικνύει τον ισχυρισμό του αποκαλύπτοντας τις δεσμεύσεις του. Οι χαμένοι αποδεικνύουν επίσης ότι οι προσφορές τους ήταν μικρότερες από την νικητήρια τιμή. Για να γίνει αποδεκτός ο νικητής, οι πλειοδότες διερευνούν πρώτα την εγκυρότητα του $y_i = g_i(\beta_i)$ και στη συνέχεια ελέγχουν αν όλα τα $n - 1$ σημεία που έχουν βρίσκονται στο $g_i(x)$.

12.4.2 Δημοπρασίες Δεύτερης Τιμής με Σφραγισμένη Προσφορά

Σε μια δημοπρασία δεύτερης τιμής με σφραγισμένη προσφορά, οι πλειοδότες υποβάλλουν στον λήπτη προσφορών μια σφραγισμένη τιμή πιθανώς της πραγματικής αποτίμησης. Οι πλειοδότες μπορούν να υποβάλλουν προσφορά εφόσον δεν έχει συμπληρωθεί η ώρα λήξης. Μόλις παρέλθει ο επιτρεπόμενος χρόνος υποβολής, οι νέες προσφορές δεν γίνονται δεκτές. Μπαίνοντας στη φάση ανοίγματος των προσφορών, ο νικητής ορίζεται ως η οντότητα με την υψηλότερη προσφορά και, όπως ορίζει ο κανόνας, ο νικητής πληρώνει τη δεύτερη υψηλότερη έγκυρη προσφορά.

Σε μια από τις πρώτες λύσεις που προτάθηκαν [46], προτείνεται η ιδέα κατασκευής μιας δημοπρασίας δεύτερης τιμής με σφραγισμένη προσφορά χρησιμοποιώντας κρυπτογραφικά πρωτόκολλα. Κατά τη φάση ανοίγματος, οι χαμένοι προσφορές παραμένουν ιδιωτικές. Μόλις καθοριστεί ένας νικητής, αποκαλύπτονται μόνο η πρώτη και η δεύτερη μεγαλύτερη προσφορά προκειμένου να καθοριστεί ο νικητής και η τιμή πώλησης. Αυτή η λύση περιλαμβάνει μια σειρά βημάτων που πρέπει να ακολουθηθούν από το πρωτόκολλο. Το πρωτόκολλο που περιγράφεται παρακάτω αφορά την περίπτωση όπου υπάρχουν μόνο δύο πλειοδότες A και B , και ένας δημοπράτης C . Κάθε πλειοδότης αναπαριστά την προσφορά του με μια τιμή στο διάστημα $[1, 100]$. Κάθε πλειοδότης προχωρά στην υποβολή των κρυπτογραφημένων προσφορών χρησιμοποιώντας το δημόσιο κλειδί του δημοπράτη και επιπλέον το δικό του δημόσιο κλειδί. Στην συνέχεια, κάθε ένας πλειοδότης υπολογίζει τη διαφορά k μεταξύ των κρυπτογραφημένων αριθμών και επιλέγει έναν αριθμό j που προσδιορίζει τη δική της προσφορά, και αποστέλλει το $k - j$ στον άλλο πλειοδότη. Ο άλλος πλειοδότης υπολογίζει μια ακολουθία τιμών με βάση την εξίσωση $y_u = D_b(k - j + u)$, όπου b το ιδιωτικό του κλειδί και u είναι όλες οι πιθανές τιμές του διαστήματος $[1, 100]$, και υπολογίζει επίσης τα $z_u = y_u \pmod{q}$, όπου q είναι ένας τυχαίος πρώτος αριθμός. Στη συνέχεια, κάθε πλειοδότης αποστέλλει την ακολουθία y_u στον άλλο πλειοδότη, από την οποία μπορεί να καταλάβει εάν η προσφορά του είναι μεγαλύτερη ή ίσως μικρότερη από την άλλη προσφορά.

Οι Naor *et al.* [47] πρότειναν ένα πρωτόκολλο δημοπρασίας σφραγισμένης προσφοράς χωρίς να υπάρχει ανάγκη ύπαρξης κάποιου αριθμού έμπιστων οντοτήτων. Το πρωτόκολλο απαιτεί τη χρήση μιας τρίτης οντότητας, που χαρακτηρίζεται ως ο εκδότης της δημοπρασίας, και η οποία είναι υπεύθυνη για την κατασκευή του λογικού κυκλώματος MPC που θα χρησιμοποιηθεί από τους δημοπράτες. Το πρωτόκολλο διασφαλίζει ότι δεν θα υπάρχει διαρροή πληροφοριών ακόμη και αν ο εκδότης της δημοπρασίας και οι δημοπράτες συνεννοηθούν μεταξύ τους. Η αποτελεσματικότητα του συγκεκριμένου πρωτοκόλλου μπορεί να βελτιωθεί με βάση μια τροποποίηση που προτάθηκε από τους Lipmaa *et al.* [48] και η οποία κάνει χρήση ενός σχήματος ομοιομορφικής κρυπτογράφησης. Οι περιορισμοί της εξαπάτησης ενός από τους διακομιστές αποτελεί θέμα συζήτησης του [49], και μια πιθανή λύση σε αυτό το πρόβλημα είναι να χωρίζεται η κάθε προσφορά σε δύο μερίδια.

Τέλος, στο [50] προτείνονται δύο πρωτόκολλα δημοπρασίας δεύτερης τιμής με σφραγισμένη προσφορά χρησιμοποιώντας τεχνικές κάλυψης (masking) και ενός σχήματος επαληθεύσιμης κοινής χρήσης μυστικών (Verifiable Secret Sharing – VSS) [51]. Σε αυτήν την προτεινόμενη λύση, το πρωτόκολλο μπορεί επίσης να

κατασκευαστεί κάνοντας χρήση ενός πιο σύνθετου σχήματος VSS των Rabin και Ben-Or [52]. Στην πρώτη λύση, οι πλειοδότες καλύπτουν (masking) τις προσφορές τους χρησιμοποιώντας την πράξη της πρόσθεσης (+) με δύο κοινά μυστικά. Ενώ στη δεύτερη λύση, οι πλειοδότες καλύπτουν τις προσφορές τους χρησιμοποιώντας δύο πράξεις, της πρόσθεσης (+) και του πολλαπλασιασμού (×), με τα δύο κοινά μυστικά. Αυτά τα πρωτόκολλα μπορούν να παρέχουν ασφάλεια τόσο με παθητικά όσο και με ενεργητικά μοντέλα αντιπάλου.

12.5 Ιδιωτική Ανάκτηση Πληροφοριών

Η ανάκτηση ιδιωτικών πληροφοριών (Private Information Retrieval – PIR) [53] παρέχει έναν τρόπο για την ανάκτηση δεδομένων από μια βάση δεδομένων χωρίς το Σύστημα Διαχείρισης Βάσεων Δεδομένων (ή ο διαχειριστής της) να μπορεί να εξάγει κάποια πληροφορία σχετικά με το συγκεκριμένο στοιχείο που ανακτήθηκε. Μια προφανής λύση για το πρόβλημα του PIR είναι να γίνει μεταφορά ολόκληρης της βάσης δεδομένων στον αιτούντα, ο οποίος στη συνέχεια κάνει τοπική ανάκτηση των στοιχείων που τον ενδιαφέρουν από τη βάση δεδομένων που έχει λάβει. Αν και αυτή η προφανής λύση προσφέρει μια τέλεια προστασία της ιδιωτικότητας, το κόστος επικοινωνίας και ο αποθηκευτικός χώρος που θα χρειαζόταν θα ήταν ασύμφορος για μεγάλες βάσεις δεδομένων.

Τα συστήματα PIR ταξινομούνται κυρίως με βάση τις εγγυήσεις ιδιωτικότητας που παρέχουν, και τον αριθμό των διακομιστών που απαιτούνται για την προστασία της ιδιωτικότητας [54]. Η εγγύηση της ιδιωτικότητας των σχημάτων PIR μπορεί να είναι είτε πληροφοριθεωρητική (information-theoretic), είτε υπολογιστική ή συνδυασμός και των δύο. Τα πληροφοριθεωρητικά σχήματα PIR (ITPIR) εγγύώνται την ιδιωτικότητα των ερωτημάτων, ανεξάρτητα από τις υπολογιστικές δυνατότητες των βάσεων δεδομένων που απαντούν στο ερώτημα του χρήστη. Εισάγοντας την έννοια του PIR το 1995, ο Chor *et al.* [53] απέδειξε ότι η προφανής λύση που αναφέρθηκε παραπάνω είναι η βέλτιστη για μια πληροφοριθεωρητική προστασία της ιδιωτικότητας κάνοντας χρήση ενός μόνο διακομιστή [53, 55]. Με άλλα λόγια, κάθε ITPIR απαιτεί απαραίτητα δύο ή περισσότερους διακομιστές που δεν συνεννοούνται (non-colluding) για να συμμετέχουν στην απάντηση του ερωτήματος. Έτσι, όλα τα σχήματα ITPIR είναι σχήματα PIR πολλών διακομιστών. Δηλαδή, ο χρήστης πρέπει να ρωτήσει πολλούς διακομιστές παράλληλα και να συνδυάσει τις απαντήσεις τους για να λάβει την απάντηση στο ερώτημά του. Τα υπολογιστικά σχήματα PIR (Computational PIR – CPIR), από την άλλη πλευρά, υποθέτουν ότι ένας υπολογιστικά περιορισμένος διακομιστής βάσεων δεδομένων δεν μπορεί να λύσει ένα δύσκολο υπολογιστικό πρόβλημα, όπως για παράδειγμα η δυσκολία παραγοντοποίησης μεγάλων ακεραίων αριθμών. Επομένως, όλα τα υπάρχοντα σχήματα PIR ενός διακομιστή είναι σχήματα CPIR. Η απαίτηση για διακομιστές που δεν συνεργάζονται στο ITPIR δεν υπάρχει πλέον στο CPIR, αλλά αυτό έχει ως αποτέλεσμα κάποιο κόστος στην αποτελεσματικότητά τους.

Κάθε σχήμα PIR αποτελείται από τρεις βασικούς αλγόριθμους: την δημιουργία ερωτημάτων, την κωδικοποίηση απάντησης και την αποκωδικοποίηση απάντησης. Για μια δεδομένη βάση δεδομένων X μεγέθους n -bits, η οποία οργανώνεται σε r μπλοκ των b -bits, ένας χρήστης που σκοπεύει να αποκρύψει το μοτίβο πρόσβασής του στο μπλοκ X_i της βάσης δεδομένων χρησιμοποιεί τον αλγόριθμο δημιουργίας ερωτήματος για να κωδικοποιήσει την ανάγκη του για την θέση i πριν στείλει στο ερώτημα στη βάση δεδομένων. Στη συνέχεια, η βάση δεδομένων χρησιμοποιεί τον αλγόριθμο κωδικοποίησης απάντησης για να συνδυάσει το ερώτημα i με κάθε καταχώρηση X_j της βάσης δεδομένων, όπου $j \in \{1, \dots, r\}$, και να επιστρέψει την κωδικοποιημένη απάντηση στον χρήστη. Τέλος, ο χρήστης κάνοντας χρήση του αλγορίθμου αποκωδικοποίησης απάντησης αποκωδικοποιεί το αποτέλεσμα που έλαβε. Στη περίπτωση μιας πληροφοριθεωρητικής προσέγγισης πολλαπλών διακομιστών, υποθέτουμε ότι υπάρχουν δύο ή περισσότερα αντίγραφα της βάσης δεδομένων και με βάση τα οποία ο χρήστης πρέπει να αλληλεπιδρά με τρόπο παρόμοιο με τον παραπάνω. Το κόστος υπολογισμού των αλγορίθμων στην πλευρά του χρήστη για τη δημιουργία ερωτημάτων και την αποκωδικοποίηση απάντησης είναι γενικά πολύ λιγότερο από ότι το κόστος του αλγορίθμου κωδικοποίησης απάντησης στην πλευρά του διακομιστή. Ένα σχήμα PIR είναι σωστό εάν επιστρέφει πάντα το σωστό μπλοκ X_i , ιδιωτικό εάν δεν διαρρέει πληροφορίες στη βάση δεδομένων σχετικά με το ερώτημα i και το μπλοκ X_i , και μη τετριμμένο εάν η πολυ-

πλοκότητα επικοινωνίας του είναι ανάλογη του n [54]. Επιπλέον, τα συμμετρικά σχήματα PIR (Symmetric PIR – SPIR) έχουν την πρόσθετη ιδιότητα ότι ο χρήστης μαθαίνει μόνο το μπλοκ X_i και δεν μαθαίνει πληροφορίες για τα άλλα μπλοκ X_j της βάσης δεδομένων, όπου $j \neq i$. Με άλλα λόγια, τα σχήματα SPIR παρέχουν απόρρητο στην βάση δεδομένων εκτός από την ιδιωτικότητα των χρηστών.

Τα σχήματα PIR μπορούν να βρουν εφαρμογές σε πολλούς και διάφορους τομείς, όπως σε βάσεις δεδομένων διπλωμάτων ευρεσιτεχνίας, τιμές μετοχών σε πραγματικό χρόνο, DNA βάσεις δεδομένων [56], καταχωρήσεις διευθύνσεων Ιστού (DNS) [57] και συμπεριφορικής ανάλυσης για δίκτυα διαφημίσεων [58]. Σε όλα τα παραπάνω σενάρια εφαρμογών, το πρόβλημα είναι ότι οι χρήστες δεν θέλουν να αποκαλύπτουν τις εναίσθητες πληροφορίες των ερωτημάτων που στέλνουν στους διάφορους διακομιστές βάσεων δεδομένων, και απαιτούν εμπιστευτικότητα των ερωτημάτων τους, τόσο ως προς τρίτους όσο και ως προς τον διακομιστή που διατηρεί τα δεδομένα που τους ενδιαφέρουν.

12.5.1 Σχήματα PIR Ενός Μόνο Διακομιστή

Αρκετά σχήματα PIR ενός μόνο διακομιστή ακολουθούν σε δομή το πρώτο σχήμα που προτάθηκε από τους Kushilevitz και Ostrovsky [59]. Κάθε ένα από τα μεταγενέστερα σχήματα που προτάθηκαν βασίζονται σε κάποια παραδοχή ενός δυσεπίλυτου υπολογιστικού προβλήματος ανάλογα με το κρυπτογραφικό σύστημα που χρησιμοποιούν. Αυτά περιλαμβάνουν σχήματα που βασίζονται σε ομαδική-ομομορφική κρυπτογράφηση [60], ευέλικτα σε μέγεθος προσθετικά ομομορφικά κρυπτοσυστήματα δημόσιου κλειδιού [61], στην φ-υπόθεση απόκρυψης (Φ-Hiding Assumption – ΦHA) [62], και πιο πρόσφατα, σε κρυπτοσυστήματα πλέγματος (lattice-based) από τους Melchor και Gaborit [63].

Για μια πιο πλήρη βιβλιογραφική ανασκόπηση των λύσεων που έχουν προταθεί για σχήματα PIR ενός διακομιστή, μπορεί κάνεις να λάβει υπόψη τις ανασκοπήσεις [64, 65]. Παρά την σχετική ευκολία της εφαρμογής των σχημάτων PIR σε μια βάση δεδομένων, το κύριο πρόβλημα που αντιμετωπίζουν αυτά τα σχήματα είναι οι υπολογιστικά δαπανηροί υπολογισμοί τους (όπως μόντουλο πράξεις πολλαπλασιασμού και ύψωσης σε δύναμη), ο αριθμός των οποίων είναι γραμμικός του μεγέθους της βάσης δεδομένων. Μια από τις σχετικά πιο αποδοτικότερες λύσεις που υπάρχουν βασίζεται σε πλέγματα (lattice-based) [63]. Σε αυτό το σχήμα η ασφάλειά του βασίζεται στο πρόβλημα του διαφορικού κρυφού πλέγματος, το οποίο αποτελεί ένα πρόβλημα NP-πλήρες (NP-complete) της θεωρίας κωδικοποίησης [66]. Επιπρόσθετα, το σχήμα αυτό κάνει χρήση Μονάδων Επεξεργασίας Γραφικών (GPUs), αντί για CPU, και αποδεικνύεται ότι με την χρήση GPU τα σχήματα PIR ενός διακομιστή μπορεί να είναι πιο αποτελεσματικά από την απλή λήψη ολόκληρης της βάσης δεδομένων από την πλευρά του χρήστη. Κάτι το οποίο επιχειρεί να καταρρίψει το ζήτημα που τίθεται από τους Sion και Carbonear [67] σχετικά με την πρακτικότητα των σχημάτων PIR ενός διακομιστή. Σε σύγκριση με άλλα σχήματα PIR ενός διακομιστή [68, 61], αυτό το σχήμα παρέχει πολύ καλύτερο ρυθμό επεξεργασίας στην πλευρά του διακομιστή, τόσο σε CPU όσο και σε GPU, της τάξης των 100 sec/GB [69], το οποίο είναι πολύ πιο αποδοτικό από το πρώτο σχήμα των Kushilevitz και Ostrovsky [59] που παρείχε έναν ρυθμό επεξεργασίας της τάξης των 10^5 sec/GB.

12.5.2 Σχήματα PIR Πολλαπλών Διακομιστών

Σε αυτή την ενότητα, για την καλύτερη κατανόηση του τρόπου λειτουργίας των πληροφοριθεωρητικών σχημάτων PIR πολλαπλών διακομιστών, παρέχεται μια επισκόπηση δύο βασικών τέτοιων σχημάτων που προτάθηκαν από τον Chor *et al.* [53] και τον Goldberg [70]. Το σχήμα του Chor [53] είναι σχετικά απλούστερο, καθώς είναι το πρώτο πρωτόκολλο PIR που προτάθηκε. Ενώ, το σχήμα του Goldberg [70] είναι ελαφρώς πιο περίπλοκο και είναι διαθέσιμος ο πηγαίος του κώδικας, κάτι το οποίο το καθιστά εξαιρετικά χρήσιμο για την πραγματοποίηση πειραμάτων απόδοσης [71, 69, 72]. Η υλοποίηση του Goldberg [70], γνωστή ως Percy++ [73], αποτελεί ένα πρότζεκτ ανοιχτού κώδικα που παρέχεται στο SourceForge. Για μια σχετικά πιο πλήρη βιβλιογραφική ανασκόπηση των λύσεων που έχουν προταθεί για σχήματα PIR πολλαπλών διακομιστών, μπορεί κάνεις να λάβει υπόψη τις εξής ανασκοπήσεις [54, 74, 75].

12.5.2.1 Σχήμα PIR του Chor

Η πολυπλοκότητα του σχετικά απλού πρωτοκόλλου του Chor [53] είναι της τάξης $O(\sqrt{n})$. Το σχήμα του Chor αποτελείται από ℓ διακομιστές που ο καθένας έχει ένα πλήρες αντίγραφο της βάσης δεδομένων. Η βάση δεδομένων D σε κάθε διακομιστή αντικατοπτρίζεται ως ένας πίνακας από bits με διαστάσεις $r \times b$, όπου η k γραμμή της κανονικής βάσης δεδομένων D αποτελεί το k μπλοκ αυτής της βάσης δεδομένων (δηλ., υπάρχουν r μπλοκ των b -bits). Κατά τη δημιουργία ερωτήματος, ο χρήστης που ενδιαφέρεται για το μπλοκ i της βάσης δεδομένων επιλέγει ℓ τυχαίες ακολουθίες από bits ρ_1, \dots, ρ_ℓ , καθεμία μήκους r , έτσι ώστε $\rho_1 \oplus \dots \oplus \rho_\ell = e_i$, όπου e_i είναι μια ακολουθία από bits μήκους r , που είναι 0 παντού εκτός από τη θέση i όπου είναι 1. Στην συνέχεια, ο χρήστης αποστέλλει το ρ_j στον διακομιστή j για κάθε $j = 1, \dots, \ell$.

Κατά την φάση της κωδικοποίησης απάντησης, κάθε διακομιστής j υπολογίζει το $R_j = \rho_j \cdot D$ πραγματοποιώντας το XOR αυτών των μπλοκ k της βάσης δεδομένων για τα οποία το k^{th} bit του ρ_j είναι 1, και επιστρέφει πίσω στον χρήστη το R_j .

Τέλος, κατά την αποκωδικοποίηση απάντησης, ο χρήστης υπολογίζει το $R_1 \oplus \dots \oplus R_\ell = (\rho_1 \oplus \dots \oplus \rho_\ell) \cdot D = e_i \cdot D$, που αποτελεί το μπλοκ i της βάσης δεδομένων. Στα παραπάνω βήματα, και για την εκτίμηση της πολυπλοκότητας $O(\sqrt{n})$ που αναφέρθηκε, αγνοούμε το πλήθος ℓ των διακομιστών επειδή είναι ένας σχετικά μικρός αριθμός.

12.5.2.2 Σχήμα PIR του Goldberg

Το σχήμα PIR πολλών διακομιστών του Goldberg [70] είναι παρόμοιο, αλλά πιο περίπλοκο από το σχήμα Chor. Η ομοιότητα έγκειται στη χρήση απλών πράξεων XOR για την πραγματοποίηση των περισσότερων υπολογισμών στην πλευρά του διακομιστή. Ωστόσο, χρησιμοποιεί την κοινή χρήση μυστικών (secret sharing) του Shamir [76] για να διαμοιράσει το διάνυσμα ερωτήματος e_i του χρήστη σε ℓ μερίδια που στη συνέχεια μεταδίδονται στους διακομιστές. Η βάση δεδομένων D ενός διακομιστή αντιμετωπίζεται ως ένας πίνακας $r \times s$ με λέξεις μεγέθους w -bits, όπου και πάλι το r είναι ο αριθμός των μπλοκ και s είναι ο αριθμός των w -bit λέξεων ανά μπλοκ. Επιπλέον, τα στοιχεία από τα οποία αποτελούνται τα e_i , ρ_j και R_j δεν είναι μεμονωμένα bits, αλλά λέξεις μεγέθους w -bits. Αυτές οι αλλαγές είναι απαραίτητες επειδή το πρωτόκολλο είναι ανθεκτικό σε ερωτήματα στα οποία οι διακομιστές μπορεί να ανταποκριθούν λανθασμένα ή να μην ανταποκριθούν καθόλου. Για λόγους απλότητας, σε αυτή την περιγραφή του σχήματος θα εστιάσουμε μόνο σε τίμιους (honest) διακομιστές, οι οποίοι ανταποκρίνονται σωστά στα ερωτήματα των χρηστών, και επιπλέον, επιλέγουμε το $w = 8$ για να απλοποιήσουμε το κόστος των υπολογισμών.

Κατά τη δημιουργία ερωτήματος, ο χρήστης κωδικοποιεί ένα ερώτημα για το μπλοκ i της βάσης δεδομένων επιλέγοντας ομοιόμορφα ℓ τυχαίες διακρίτες μη μηδενικές τιμές $\alpha_1, \dots, \alpha_\ell$ μεγέθους 8-bits. Στη συνέχεια, ο χρήστης επιλέγει r πολυώνυμα βαθμού t , ένα για κάθε μπλοκ της βάσης δεδομένων D . Οι συντελεστές των μη-σταθερών όρων για το πολυώνυμο f_j είναι τυχαία στοιχεία μεγέθους 8-bits, ενώ οι συντελεστές των σταθερών όρων είναι 1 εάν $i = j$ και 0 σε διαφορετική περίπτωση. Έπειτα, ο χρήστης αποστέλλει σε κάθε διακομιστή j ένα διάνυσμα ρ_j που διαμορφώνεται από τον υπολογισμό των τιμών όλων των r πολυωνύμων για την τιμή α_j , δηλαδή $\rho_j = [f_1(\alpha_j), \dots, f_r(\alpha_j)]$ (σημειώνεται ότι κάθε $f_k(\alpha_j)$ είναι ένα στοιχείο μεγέθους 8-bits).

Αντίστοιχα, και με τρόπο παρόμοιο με το σχήμα Chor, γίνεται η κωδικοποίηση και αποκωδικοποίηση απάντησης. Πιο αναλυτικά, κάθε διακομιστής υπολογίζει ένα διάνυσμα απάντησης $R_j = \rho_j \cdot D$, όπου καθένα από τα s στοιχεία του διανύσματος $R_j = [r_{j1}, \dots, r_{js}]$ έχουν επίσης μέγεθος 8-bits. Οι διακομιστές στέλνουν το διάνυσμα R_j στον χρήστη και αυτός υπολογίζει το αποτέλεσμα του ερωτήματος χρησιμοποιώντας την παρεμβολή (interpolation) Lagrange και για το πολυώνυμο που προκύπτει υπολογίζει την τιμή του πολυωνύμου στο σημείο $x = 0$ και η τιμή αυτή αποτελεί το επιθυμητό μπλοκ της βάσης δεδομένων. Συγκεκριμένα, ο χρήστης λαμβάνει $[r_{11}, \dots, r_{1s}], \dots, [r_{\ell 1}, \dots, r_{\ell s}]$ από τους ℓ διακομιστές. Το μπλοκ i της βάσης δεδομένων υπολογίζεται αξιολογώντας για το σημείο $x = 0$ ένα διάνυσμα από s μοναδικά πολυώνυμα, το καθένα βαθμού t , δηλαδή $[\phi_1(0), \dots, \phi_s(0)]$. Αυτό επιτυγχάνεται υπολογίζοντας την παρεμβολή Lagrange για κάθε μοναδικό πολυώνυμο ϕ_i που περνά από τα ℓ σημεία $(\alpha_1, r_{1i}), (\alpha_2, r_{2i}), \dots, (\alpha_\ell, r_{\ell i})$.

Τέλος, το σχήμα PIR του Goldberg [70] παρέχει καλή υποστήριξη για την ανθεκτικότητα των ερωτημάτων έναντι διακομιστών που συνεργάζονται μεταξύ τους. Πιο αναλυτικά, σε ένα σενάριο όπου οι χρήστες υποβάλλουν τα ερωτήματά τους σε τουλάχιστον k από τους ℓ διακομιστές, το σύστημα μπορεί να ανεχτεί έως και ℓ κακόβουλους διακομιστές που απαντούν λανθασμένα (όπου $u < k - \lfloor \sqrt{k}t \rfloor$ και ο συμβολισμός $\lfloor x \rfloor$ υποδηλώνει τον μεγαλύτερο ακέραιο αριθμό μικρότερο ή ίσο του x), χωρίς να εμποδίζει την ικανότητα των χρηστών να ανακτήσουν τη σωστή απάντηση, και t διακομιστές που συνεργάζονται μεταξύ τους (όπου $0 < t < k \leq \ell$), χωρίς να διακυβεύεται η ιδιωτικότητα των ερωτημάτων των χρηστών.

12.6 Εξόρυξη Δεδομένων με Διασφάλιση Ιδιωτικότητας

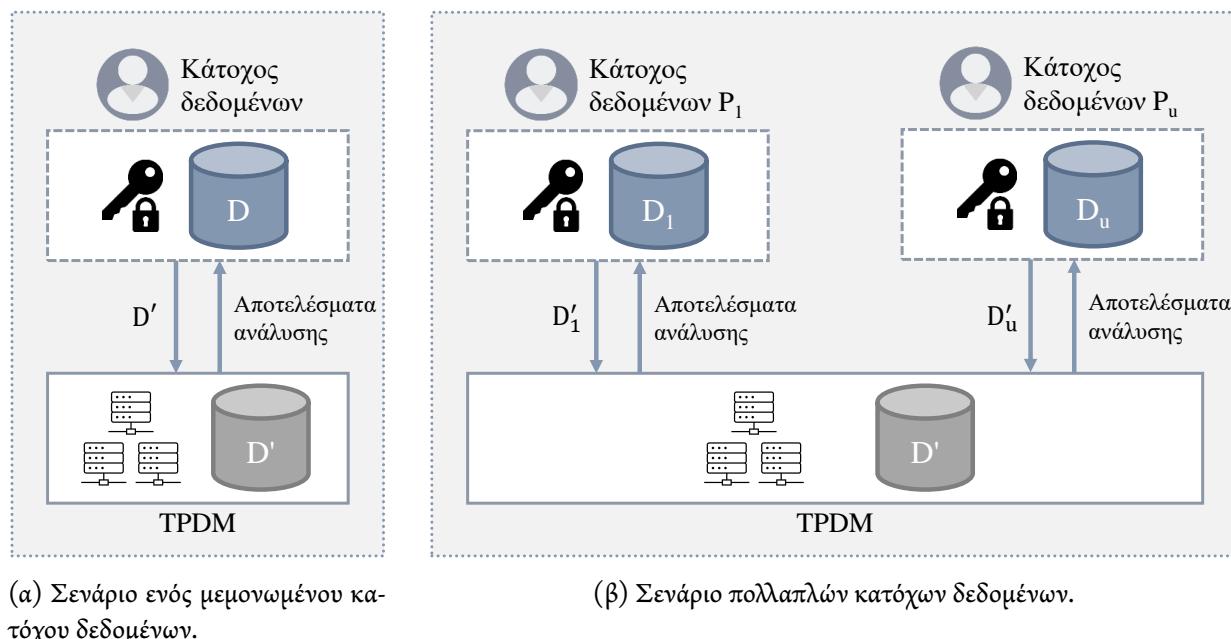
Η εξόρυξη δεδομένων (data mining) αποτελεί την διαδικασία εξαγωγής και ανακάλυψης προτύπων (patterns) σε μεγάλα σύνολα δεδομένων αξιοποιώντας μεθόδους από διάφορους τομείς, όπως την μηχανική μάθηση, την στατιστική και τα συστήματα βάσεων δεδομένων. Η εξόρυξη δεδομένων συνιστά ένα διεπιστημονικό υποπεδίο της επιστήμης των υπολογιστών και της στατιστικής με γενικό στόχο την εξαγωγή πληροφοριών (με «έξυπνες» μεθόδους) από ένα σύνολο δεδομένων και τη μετατροπή αυτής της πληροφορίας σε μια κατανοητή δομή για περαιτέρω χρήση [77]. Επιπλέον, η εξόρυξη δεδομένων αποτελεί ένα βήμα ανάλυσης από την διαδικασία «ανακάλυψης γνώσης σε βάσεις δεδομένων» (Knowledge Discovery in Databases – KDD) [78]. Εκτός από το βήμα ανάλυσης των πρωτογενών δεδομένων, περιλαμβάνει επίσης πτυχές διαχείρισης βάσεων δεδομένων και δεδομένων, προ-επεξεργασία δεδομένων, εκτιμήσεις μοντέλων και συμπερασμάτων, μετρικές ενδιαφέροντος, εκτιμήσεις πολυπλοκότητας και μετα-επεξεργασία των δομών που ανακαλύφθηκαν. Η εξόρυξη δεδομένων πραγματοποιεί ημιαυτόματη ή αυτόματη ανάλυση μεγάλων ποσοτήτων δεδομένων για την εξαγωγή προηγουμένως άγνωστων, ενδιαφέροντων προτύπων, όπως ομάδες δεδομένων (συσταδοποίηση - clustering), γενίκευση γνωστών δομών με εφαρμογή τους σε νέα δεδομένα (ταξινόμηση - classification), ασυνήθιστες εγγραφές (ανίχνευση ανωμαλιών) και εξαρτήσεις (εξόρυξη κανόνων συσχέτισης και εξόρυξη διαδοχικών προτύπων).

Στην εξόρυξη δεδομένων, η διασφάλιση της ιδιωτικότητας αποτελεί ένα σοβαρό πρόβλημα, ειδικά σε εφαρμογές που περιλαμβάνουν τη συλλογή και την κοινή χρήση προσωπικών δεδομένων. Το πρόβλημα της εξόρυξης δεδομένων με διασφάλιση ιδιωτικότητας (Privacy Preserving Data Mining – PPDM) αποτελεί ένα ζήτημα προεξέχουσας σημασίας λόγω της αυξανόμενης ανάγκης των επιχειρήσεων να αναθέτουν σε εξωτερικούς συνεργάτες την ανάλυση των δεδομένων ή/και την από κοινού/συνεργατική ανάλυση αυτών. Κατά συνέπεια, έχουν προταθεί πολλές τεχνικές που έχουν σχεδιαστεί για τη διασφάλιση της ιδιωτικότητας ή του απορρήτου των δεδομένων (ανάλογα με την φύση των δεδομένων), ενώ ταυτόχρονα διατηρούν τη χρησιμότητα των δεδομένων, όταν η ανάλυση δεδομένων πρέπει να διεξάγεται με ασφάλεια σε συνεργατικό περιβάλλον. Ορισμένες από αυτές τις τεχνικές απευθύνονται στο σενάριο ενός μεμονωμένου κατόχου δεδομένων, ενώ άλλες εφαρμόζονται στο σενάριο πολλαπλών κατόχων δεδομένων [79]. Αυτά τα δύο σενάρια είναι επίσης γνωστά ως μεμονωμένη εξόρυξη/ανάλυση δεδομένων και συνεργατική/κοινή εξόρυξη/ανάλυση δεδομένων, αντίστοιχα.

Το σενάριο ενός μεμονωμένου κατόχου δεδομένων (single data owner) αναφέρεται στην περίπτωση όπου ένας μεμονωμένος κάτοχος δεδομένων αναθέτει σε έναν τρίτο αναλυτή δεδομένων (Third Party Data Miner – TPDM) την ανάλυση του ιδιωτικού του συνόλου δεδομένων D . Αυτό το σενάριο αποτυπώνεται στο Σχήμα 12.9a. Ο στόχος μιας τέτοιας ανάθεσης είναι η μείωση του λειτουργικού κόστους για την διεξαγωγή της εξόρυξης δεδομένων, αξιοποιώντας την υπολογιστική ισχύ που είναι διαθέσιμη στο TPDM (συνήθως ένας πάροχος υπολογιστικού νέφους). Ο κάτοχος δεδομένων δίνει εντολή στο TPDM να εκτελέσει την επιθυμητή ανάλυση δεδομένων για λογαριασμό του, πάνω στα δεδομένα που του ανατέθηκαν, καθορίζοντας τη φύση της επιθυμητής ανάλυσης και τις σχετικές παραμέτρους. Το TPDM εκτελεί την επιθυμητή ανάλυση και στη συνέχεια στέλνει τα αποτελέσματα πίσω στον κάτοχο των δεδομένων. Οι ιδανικές απαιτήσεις διασφάλισης της ιδιωτικότητας σε αυτήν την περίπτωση, κάνοντας χρήση κρυπτογραφικών τεχνικών, είναι οι εξής [79]:

1. Το TPDM δεν πρέπει να έχει άμεση πρόσβαση στο μη κρυπτογραφημένο σύνολο δεδομένων.

2. Τυχόν ενδιάμεσα αποτελέσματα, όπως κεντροειδή (centroids) συστάδας, πραγματικές (σημασιολογικές) αποστάσεις μεταξύ εγγραφών, αποτελέσματα συναρτήσεων ενεργοποίησης (activation functions) νευρωνικών δικτύων και βάρη νευρωνικών δικτύων, δεν πρέπει να αποκαλύπτονται σε καμία περίπτωση στο TPDM.
3. Ο κάτοχος των δεδομένων είναι ο μόνος που έχει πρόσβαση στα αποτελέσματα εξόρυξης δεδομένων ή/και στο μοντέλο που προέκυψε.



Σχήμα 12.9: Πιθανά σενάρια κατόχων δεδομένων στην εξόρυξη δεδομένων.

Από την άλλη πλευρά, το σενάριο πολλαπλών κατόχων δεδομένων (multiple data owners) αναφέρεται στην περίπτωση όπου τα δεδομένα $D = \{D_1, \dots, D_u\}$ κατανέμονται σε πολλαπλούς κατόχους δεδομένων $P = \{p_1, \dots, p_u\}$ οι οποίοι επιθυμούν να μοιραστούν τα δεδομένα τους στο πλαίσιο κάποιας συνεργατικής εξόρυξης δεδομένων, για παράδειγμα την δημιουργία ενός κοινού μοντέλου ταξινόμησης ή συσταδοποίησης. Αυτό το σενάριο απεικονίζεται στο Σχήμα 12.9β. Σε αυτό το σενάριο υποστηρίζεται ότι με την κοινή χρήση δεδομένων παράγεται ένα καλύτερο αποτέλεσμα εξόρυξης δεδομένων από αυτό που θα μπορούσε να είχε προκύψει εάν χρησιμοποιούνταν μόνο τα τοπικά δεδομένα ενός μόνο κατόχου. Όσον αφορά την ιδιωτικότητα των δεδομένων, η συνεργατική εξόρυξη δεδομένων πρέπει σαφώς να διεξάγεται με ασφαλή τρόπο [80]. Αντίστοιχα, οι ιδιαίκες απαιτήσεις διασφάλισης της ιδιωτικότητας σε αυτήν την περίπτωση, κάνοντας χρήση κρυπτογραφικών τεχνικών, είναι οι εξής [79]:

1. Η ιδιωτικότητα των δεδομένων που ανήκουν σε κάθε συμμετέχοντα p_i πρέπει να διασφαλίζεται σε σχέση με το TPDM και σε σχέση με όλους τους υπόλοιπους συμμετέχοντες.
2. Τα ενδιάμεσα αποτελέσματα που παράγονται κατά την εξόρυξη δεδομένων δεν πρέπει να αποκαλύπτονται στο TPDM ή σε οποιονδήποτε άλλο συμμετέχοντα.
3. Κάθε συμμετέχων λαμβάνει τα συνολικά αποτελέσματα ανάλυσης δεδομένων ή τα αποτελέσματα που αφορούν μόνο τα δικά του δεδομένα (και σε καμία περίπτωση τα αποτελέσματα των άλλων συμμετεχόντων).

Στις υποενότητες που ακολουθούν, παρουσιάζεται μια ανασκόπηση των υφιστάμενων τεχνικών κρυπτογράφησης που χρησιμοποιούνται στην εξόρυξη δεδομένων για την διασφάλιση της ιδιωτικότητας. Αυτές οι

τεχνικές ομαδοποιούνται με βάση μια από τις ακόλουθες κύριες κρυπτογραφικές τεχνικές: (α) ασφαλείς υπολογισμοί πολλαπλών οντοτήτων (MPC), (β) κοινή χρήση μυστικών (secret sharing), και (γ) ομοιορφική κρυπτογράφηση.

12.6.1 Λύσεις PPDM με Χρήση Ασφαλών Υπολογισμών Πολλαπλών Οντοτήτων

Οι ασφαλείς υπολογισμοί πολλαπλών οντοτήτων (Secure Multi-Party Computation – MPC) αποτελούν ένα καθιερωμένο ερευνητικό πεδίο που περιλαμβάνει ένα σύνολο πρωτοκόλλων τα οποία επιτρέπουν σε έναν αριθμό οντοτήτων να υπολογίζουν συνεργατικά τα αποτελέσματα κάποιων συναρτήσεων ή απλά κάποιων στατιστικών στοιχείων με βάση τα δεδομένα που κατέχουν και χωρίς να αποκαλύπτουν το τι δεδομένα έδωσαν ως είσοδο (βλέπε Ενότητα 8.1). Υπάρχουν αρκετά πρωτόκολλα MPC που έχουν σχεδιαστεί για να υποστηρίζουν τον ασφαλή υπολογισμό διαφόρων συναρτήσεων, όπως το άθροισμα [81], ο πολλαπλασιασμός [82], το εσωτερικό γινόμενο [83, 82, 84] και η σύγκριση δεδομένων [39]. Οι τεχνικές ασφαλείας που χρησιμοποιούνται για την εγγύηση της ασφάλειας των υποκείμενων υπολογισμών εκτελούνται από κοινού μεταξύ των συμμετεχόντων και ανάλογα με τη φύση του εκάστοτε πρωτοκόλλου MPC.

Η χρήση ενός πρωτοκόλλου MPC προτάθηκε για πρώτη φορά από τους Lindell και Pinkas [85] ως μια πιθανή λύση για την ασφάλεια και την ιδιωτικότητα του PPDM, με έμφαση στην ασφαλή υλοποίηση του ταξινομητή δέντρου αποφάσεων ID3. Στη συνέχεια, έχουν προταθεί πολλά πρωτόκολλα MPC που αποσκοπούν στην υλοποίηση θεμελιωδών συναρτήσεων που απαιτούνται από τους διάφορους αλγόριθμους εξόρυξης δεδομένων. Χρησιμοποιώντας αυτά τα πρωτόκολλα, έχουν υλοποιηθεί μια σειρά από ασφαλείς αλγόριθμους εξόρυξης δεδομένων. Παραδείγματα αυτών περιλαμβάνουν τους DBSCAN [83, 82, 86, 81, 87], k-Means [88] και NNC (Nearest Neighbor Clustering) [84]. Αυτές οι υλοποίησεις έλαβαν υπόψη τους διαφορετικούς αριθμούς συμμετεχόντων, όπως δύο οντοτήτων [83, 86, 81, 84] και πολλαπλών οντοτήτων [88, 82], αλλά και διαφορετικές κατατμήσεις δεδομένων, όπως οριζόντια [88, 83, 82, 86, 84], κάθετη [83, 86, 81] και αυθαίρετη [86]. Επιπλέον, χρησιμοποιήθηκαν επίσης μια σειρά μηχανισμών για τον προσδιορισμό της «ομοιότητας» (similarity) μεταξύ των διαφορετικών εγγραφών δεδομένων οι οποίες κατανέμονται σε πολλαπλούς κατόχους δεδομένων: (i) το ασφαλές εσωτερικό γινόμενο [83, 82, 84], (ii) τον ασφαλή υπολογισμό αθροισμάτων [81], και (iii) την ασφαλή σύγκριση [83, 82, 86, 81, 84] κάνοντας χρήση του πρωτοκόλλου του Yao [89]. Σε όλες αυτές τις προτεινόμενες λύσεις, οι ίδιοι οι κάτοχοι των δεδομένων αναλαμβάνουν ένα σημαντικό μέρος των υπολογισμών και ως εκ τούτου απαιτείται να διαθέτουν επαρκείς πληροφοριακούς πόρους και την ικανότητα για τη διεξαγωγή της επιθυμητής ανάλυσης. Επομένως, η χρήση των πρωτοκόλλων MPC, ανεξάρτητα από το ακριβές πρωτόκολλο που υιοθετήθηκε, εισάγει ένα γενικό κόστος υπολογισμού και επικοινωνίας από την πλευρά των κατόχων δεδομένων, καθιστώντας την προσέγγιση εφαρμόσιμη μόνο για μικρά σύνολα δεδομένων και μικρό αριθμό κατόχων δεδομένων.

Επιπρόσθετα, υπάρχουν έρευνες που προσπαθούν να μειώσουν αυτές τις επιβαρύνσεις των υπολογισμών και της επικοινωνίας στα πρωτόκολλα MPC εισάγοντας μια τρίτη οντότητα που ενεργεί ως διαμεσολαβητής στον ασφαλή υπολογισμό [90, 88, 83, 82]. Αυτή η τρίτη οντότητα, στις λύσεις που υπάρχουν, θεωρείται ότι είναι έμπιστη ή ημι-έντιμη. Μια έμπιστη τρίτη οντότητα (Trusted Third Party – TTP) συμπεριφέρεται «τίμια», επομένως δεν θα παρεκκλίνει από την προσχεδιασμένη διαδικασία και δεν θα χρησιμοποιήσει ενδιάμεσα αποτελέσματα υπολογισμών για περαιτέρω έρευνα, ενώ μια ημι-έντιμη τρίτη οντότητα (Semi-honest Third Party – STP) θα έχει τίμια συμπεριφορά κατά την εκτέλεση της προσχεδιασμένης διαδικασίας, ωστόσο τα ενδιάμεσα αποτελέσματα υπολογισμών μπορεί να αναλυθούν για την ανάκτηση πρόσθετων πληροφοριών. Ως εκ τούτου, η πρώτη περίπτωση δεν απαιτεί την υιοθέτηση κάποιων μέτρων ασφαλείας, ενώ η δεύτερη σίγουρα απαιτεί κάποια επιπρόσθετα μέτρα ασφαλείας για την αποφυγή της εξαγωγής συμπερασμάτων χρησιμοποιώντας τα ενδιάμεσα αποτελέσματα υπολογισμών. Στο [83] χρησιμοποιήθηκε μια οντότητα TTP για τον υπολογισμό των συνολικών στατιστικών στοιχείων για λογαριασμό των κατόχων δεδομένων, έχοντας ως αποτέλεσμα την μείωση της πολυπλοκότητας επικοινωνίας του προσαρμοσμένου πρωτοκόλλου MPC. Ωστόσο, σε αυτή την περίπτωση οι κάτοχοι δεδομένων εξακολουθούν να χρειάζεται να εκτελέσουν ολόκληρη την εξόρυξη δεδομένων και έτσι δεν αποφεύγεται η υπολογιστική πολυπλοκότητα. Το πιο σημαντικό σε αυτές τις

λύσεις είναι ότι η εμπλοκή μιας οντότητας TTP προκαλεί ανησυχία και για πολλούς κατόχους δεδομένων αποτελεί κίνδυνο ασφάλειας. Αυτό το ζήτημα ασφαλείας αντιμετωπίστηκε στο [88] με τη συμμετοχή μιας οντότητας STP που είχε πρόσβαση μόνο σε κρυπτογραφημένα δεδομένα.

Ένα πρωτόκολλο MPC παρέχει μια σταθερή θεωρητική βάση για το PPDM και η ασφάλεια του μπορεί να αποδειχθεί χρησιμοποιώντας αποδείξεις μηδενικής γνώσης (Zero Knowledge Proofs – ZKP), οι οποίες με την σειρά τους διασφαλίζουν ότι κάθε συμβαλλόμενο μέρος έχει αποκτήσει «μηδενική» γνώση σχετικά με τα δεδομένα και τα αποτελέσματα άλλων συμμετεχόντων [91]. Ωστόσο, στην περίπτωση που τα ενδιάμεσα αποτελέσματα ενός αλγόριθμου εξόρυξης δεδομένων αποκαλύπτονται σε όλους τους συμμετέχοντες, αυτό δημιουργεί μια ευπάθεια ασφαλείας, ειδικά όταν εμπλέκεται ένας μη-τίμιος κάτοχος δεδομένων. Ένας μη-τίμιος συμμετέχων μπορεί να πραγματοποίησει μια επίθεση επικάλυψης (overlapping attack) και να χρησιμοποιήσει τα ληφθέντα αποτελέσματα υπολογισμών για την εύρεση ομοιοτήτων μεταξύ των δεδομένων που ανήκουν σε άλλους συμμετέχοντες και στα δικά του δεδομένα [86, 87].

12.6.2 Λύσεις PPDM με Κοινή Χρήση Μυστικών

Η ιδέα της κοινής χρήσης μυστικών (secret sharing) εισήχθηκε για πρώτη φορά ως μια κρυπτογραφική μέθοδος στις εργασίες [76, 92]. Αυτή η μέθοδος περιλαμβάνει τη χρήση ενός σχήματος κρυπτογράφησης με τέτοιο τρόπο ώστε το μυστικό κλειδί SK να χωρίζεται σε έναν αριθμό μεριδίων $\{sk_1, sk_2, \dots, sk_u\}$ που διανέμονται σε ένα προκαθορισμένο σύνολο u συνεργαζόμενων συμμετεχόντων. Κάθε συμμετέχων μπορεί να κρυπτογραφεί τις εγγραφές δεδομένων του ανεξάρτητα από τους άλλους συμμετέχοντες χρησιμοποιώντας το δημόσιο κλειδί του συστήματος. Ωστόσο, η αποκρυπτογράφηση των δεδομένων είναι αρκετά πιο σύνθετη. Το μυστικό κλειδί SK , που χρησιμοποιείται για την αποκρυπτογράφηση των δεδομένων, θα πρέπει να ανακατασκευαστεί συνδυάζοντας τα μερίδια που ανήκουν στα συμμετέχοντα μέρη. Για παράδειγμα, σε σχήματα κατωφλίου t , τα κρυπτοκείμενα μπορούν να αποκρυπτογραφηθούν συνδυάζοντας οποιαδήποτε μερίδια των t ή περισσότερων συμμετεχόντων. Ωστόσο, κάτω από αυτό το κατώφλι μεριδίων δεν μπορεί να προκύψει μια έγκυρη αποκρυπτογράφηση. Επιπλέον, υπάρχουν διάφορα σχήματα κρυπτογράφησης που έχουν σχεδιαστεί για να λειτουργούν χρησιμοποιώντας την έννοια της κοινής χρήσης μυστικών, που επεκτείνονται από τυπικές μορφές κρυπτογράφησης, όπως στη λύση [93], έως το σχήμα PPE (Property Preserving Encryption) [94], όπως στη λύση [95]. Τα σχήματα PPE που υποστηρίζουν την κοινή χρήση μυστικών μπορούν να προσφέρουν μια νέα μορφή συνεργατικής εξόρυξης δεδομένων, όπου οι συμμετέχοντες μπορούν να κρυπτογραφούν τοπικά τα δεδομένα τους, ενώ στη συνέχεια αυτά αποστέλλονται σε έναν τρίτο αναλυτή δεδομένων (Third Party Data Miner – TPDM) ο οποίος χρησιμοποιεί τις ιδιότητες του σχήματος PPE για να χειριστεί ολόκληρο το σύνολο δεδομένων για λογαριασμό των κατόχων δεδομένων.

Αν και η συμμετοχή των κατόχων δεδομένων μειώνεται (υπολογιστικά και επικοινωνιακά) χρησιμοποιώντας την έννοια της κοινής χρήσης μυστικών σε σύγκριση με αυτήν που υπάρχει στα πρωτόκολλα MPC, ωστόσο η κοινή χρήση μυστικών παρουσιάσει ορισμένους περιορισμούς. Πρώτον, η απαίτηση να είναι οι συμμετέχοντες ημι-έντιμοι και να μην συνεννοούνται μεταξύ τους, αποτελεί μια σημαντική ανησυχία για αυτού του τύπου τα σχήματα, ενώ για πολλούς κατόχους δεδομένων αποτελεί κίνδυνο για την ασφάλεια. Καθώς η κοινή χρήση μυστικών επιτρέπει την αποκρυπτογράφηση δεδομένων κάθε φορά που συνδυάζονται t μερίδια του μυστικού κλειδιού, αυτό μπορεί να έχει ως αποτέλεσμα οι συμμετέχοντες που συνεννοούνται (colluding) μεταξύ τους, να μπορούν να αποκρυπτογραφήσουν τα δεδομένα χωρίς την άδεια του κατόχου των δεδομένων. Δεύτερον, η κοινή χρήση μυστικών τείνει να είναι αναποτελεσματική μόνο για μεγάλα σύνολα δεδομένων και επομένως είναι κατάλληλη για ένα περιορισμένο σύνολο της συνεργατικής εξόρυξης δεδομένων. Τρίτον, καθώς το μυστικό κλειδί κατανέμεται μεταξύ των συμμετεχόντων, οι κάτοχοι δεδομένων δεν μπορούν να αποκρυπτογραφήσουν τα δικά τους δεδομένα χωρίς τη συμμετοχή των άλλων μερών, και επομένως ένα αντίγραφο των ιδιωτικών δεδομένων πρέπει να διαφυλάσσεται και τοπικά από κάθε μεμονωμένο κάτοχο δεδομένων.

12.6.3 Λύσεις PPDM με Χρήση Ομομορφικής Κρυπτογράφησης

Χρησιμοποιώντας ένα σχήμα ομομορφικής κρυπτογράφησης (βλέπε Ενότητα 8.2) η ιδιωτικότητα των δεδομένων διασφαλίζεται κρυπτογραφώντας το σύνολο δεδομένων προτού αυτά ανατεθούν σε έναν τρίτο αναλυτή δεδομένων (TPDM), έτσι ώστε αυτός να μην έχει πρόσβαση στα δεδομένα σε μορφή απλού κειμένου και φυσικά να μην διαθέτει επίσης πρόσβαση στο σχετικό κλειδί αποκρυπτογράφησης. Στη συνέχεια, ο TPDM χειρίζεται τα κρυπτογραφημένα δεδομένα χρησιμοποιώντας τις ομομορφικές ιδιότητες του εκάστοτε σχήματος ομομορφικής κρυπτογράφησης. Πολλοί αλγόριθμοι ασφαλούς εξόρυξης δεδομένων έχουν υλοποιηθεί χρησιμοποιώντας αυτήν την τεχνική, συμπεριλαμβανομένων των αλγορίθμων, όπως ο k-Means [90, 96, 97], ο DBSCAN [98], η εξόρυξη κανόνων συσχέτισης [99], ο k-NN [100] και νευρωνικά δίκτυα BPNN (Back Propagation Neural Network) [101]. Ωστόσο, όπως έχει ήδη σημειωθεί, τα σχήματα ομομορφικής κρυπτογράφησης υποστηρίζουν μόνο περιορισμένες λειτουργίες και επομένως όταν απαιτούνται μη υποστηριζόμενες λειτουργίες από τον εκάστοτε αλγόριθμο εξόρυξης δεδομένων πρέπει να εφαρμόζονται εναλλακτικές μέθοδοι.

Η πιο κοινή προσέγγιση που χρησιμοποιείται για την αντιμετώπιση των περιορισμένων λειτουργιών των σχημάτων ομομορφικής κρυπτογράφησης είναι και πάλι η πραγματοποίηση των απαιτούμενων λειτουργιών από τους ίδιους τους κατόχους δεδομένων πραγματοποιώντας τις όποιες πράξεις σε μη κρυπτογραφημένα δεδομένα, και τα αποτελέσματα αυτών, στην συνέχεια, να κρυπτογραφούνται και να αποστέλλονται πίσω στο TPDM [98, 102, 90]. Για παράδειγμα, στο [90] όπου εξετάζεται ένας ασφαλής μηχανισμός για το k-Means, ο υπολογισμός των αποστάσεων μεταξύ των διαφορετικών τιμών δεδομένων αλλά και των κεντροειδών συστάδας, που βασίζονται στις ιδιότητες της ομομορφικής κρυπτογράφησης, ανατίθεται σε κάθε επανάληψη του k-Means στον κάτοχο δεδομένων για να καθορίζει την κατάλληλη συστάδα (cluster) κάθε κρυπτογραφημένης εγγραφής. Η πολυπλοκότητα της συμμετοχής του κατόχου δεδομένων σε αυτήν την περίπτωση είναι $O(k \times n \times i)$, όπου τα k , n και i είναι ο επιθυμητός αριθμός συστάδων που θα δημιουργηθούν χρησιμοποιώντας το k-Means, ο αριθμός των εγγραφών του σύνολο δεδομένων, και ο αριθμός των επαναλήψεων, αντίστοιχα.

Το κύριο πλεονέκτημα της λύσεων ομομορφικής κρυπτογράφησης είναι ότι κανένα μέρος, εκτός από τον κάτοχο δεδομένων, δεν έχει πρόσβαση σε ενδιάμεσα αποτελέσματα όταν βρίσκεται σε εξέλιξη ο αλγόριθμος εξόρυξης. Ωστόσο, η ποσότητα και η πολυπλοκότητα της ανάθεσης λειτουργιών στους κατόχους δεδομένων εισάγει ένα ανεπιθύμητο κόστος επικοινωνίας και υπολογισμού εκ μέρους των κατόχων δεδομένων. Το πιο σημαντικό, στο πλαίσιο της συνεργατικής εξόρυξης δεδομένων, είναι ότι η συμμετοχή του κατόχου δεδομένων στη σύγκριση αποστάσεων ανάμεσα σε δεδομένα που βρίσκονται μεταξύ των συμμετεχόντων αυξάνει τη πιθανότητα μιας επίθεσης επικάλυψης (overlapping) όπου ένας συμμετέχων μπορεί να εκτιμήσει τις τιμές των δεδομένων που ανήκουν σε άλλους κατόχους δεδομένων χρησιμοποιώντας γνώση των δικών του δεδομένων και των αποτελεσμάτων συγκρίσης [86]. Τέτοιες επιθέσεις μπορούν να αποτραπούν με την υιοθέτηση καθιερωμένων πρωτοκόλλων MPC για τη σύγκριση τιμών, όπως το κλασικό σχήμα του Yao [89] και το σχήμα του Cachin [39]. Ωστόσο, αυτά έχουν εξίσου σημαντικά μειονεκτήματα με αυτά που συναντώνται κατά τη χρήση πρωτοκόλλων MPC.

Για να μειωθεί η ανάγκη συμμετοχής των κατόχων δεδομένων στους υπολογισμούς, όταν ένα σύνολο δεδομένων μοιράζεται σε πολλά μέρη, θα πρέπει ο έλεγχος της ομοιότητας μεταξύ των εγγραφών να γίνεται αναφορικά σε έναν τυχαία επιλεγμένο κάτοχο δεδομένων, όπως προτείνεται στο [90]. Επιπλέον, για να αποφευχθεί η συμμετοχή του κατόχου δεδομένων στους υπολογισμούς, θα μπορούσε να γίνει εισαγωγή τρίτων οντοτήτων που βρίσκονται στο Νέφος και εκπροσωπούν τους συμμετέχοντες, όπως οι λύσεις [96, 103, 97]. Η κύρια ιδέα πίσω από αυτή την προσέγγιση είναι ότι τα δεδομένα προς ανάλυση βρίσκονται σε μια υπηρεσία Νέφους και το μυστικό κλειδί σε μια άλλη υπηρεσία Νέφους, ενώ η δεύτερη υπηρεσία ενεργεί για λογαριασμό των κατόχων δεδομένων όποτε απαιτούνται μη υποστηριζόμενες λειτουργίες από το ομομορφικό πρωτόκολλο. Η υπηρεσία Νέφους που κρατά το μυστικό κλειδί μπορεί να αποκρυπτογράφήσει τα ενδιάμεσα αποτελέσματα, να εκτελέσει τις απαιτούμενες λειτουργίες σε απλό κείμενο και να στείλει τα κρυπτογραφημένα αποτελέσματα πίσω στην υπηρεσία Νέφους που εκτελεί την ανάλυση δεδομένων. Ωστόσο, αυτή η προσέγγιση αυξάνει το κόστος της εξωτερικής ανάθεσης των δεδομένων και βασίζεται στην υπόθεση ότι οι πάροχοι υπηρεσιών Νέφους

δεν συνεννοούνται (non-colluding) μεταξύ τους.

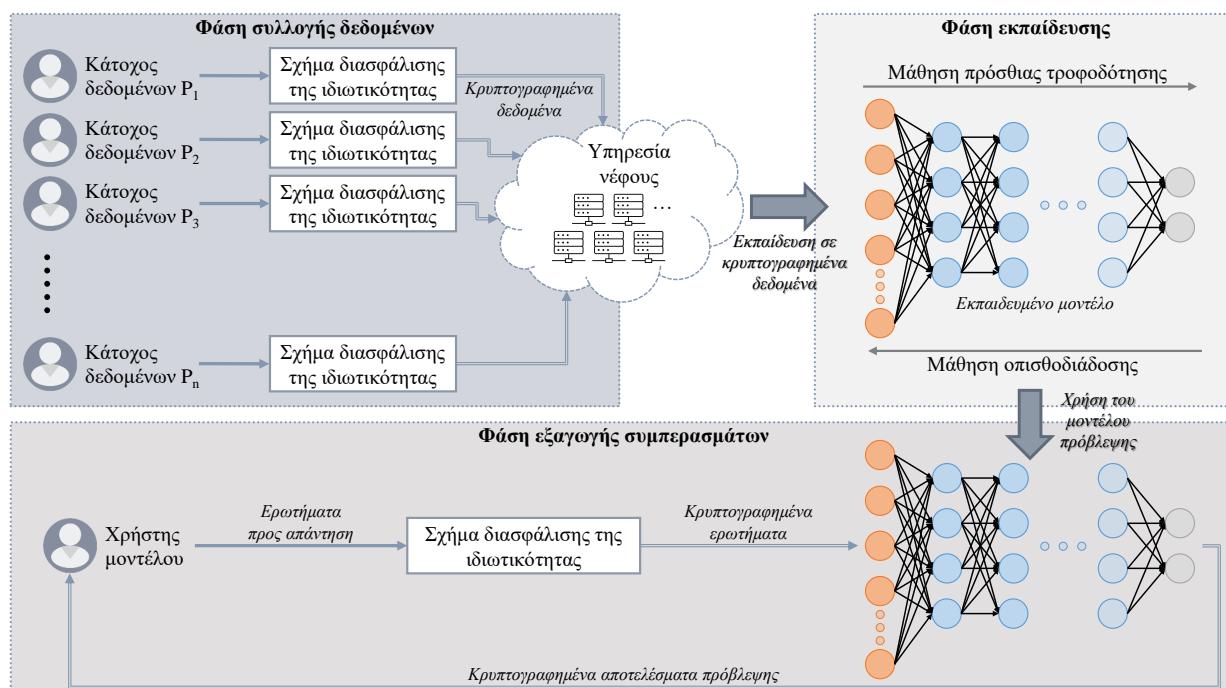
12.7 Μηχανική Μάθηση με Διασφάλιση Ιδιωτικότητας

Η μηχανική μάθηση (Machine Learning – ML) ορίζεται ως ο σχεδιασμός ενός συστήματος που μπορεί να μαθαίνει και να επιλύει προβλήματα κάνοντας χρήση γνώσης από το περιβάλλον του. Οι πρόσφατες εξελίξεις στην πληροφορική παρέχουν την δυνατότητα σε πολλές εταιρείες να βασίζονται όλο και περισσότερο στα δεδομένα και να συλλέγουν/μοιράζονται/επεξεργάζονται όλο και περισσότερες πληροφορίες σχετικά για τους πελάτες τους. Για την επεξεργασία αυτών των δεδομένων, χρησιμοποιούνται τεχνικές μηχανικής μάθησης, γεγονός που καθιστά τη μηχανική μάθηση όλο και περισσότερο πολύτιμη. Η αξιοποίηση των τεχνικών μηχανικής μάθησης διασφαλίζει την εύκολη χρήση της πληθώρας των δεδομένων που προέρχονται από τους χρήστες για τον σχεδιασμό συστημάτων που είναι ικανά να επιλύουν προβλήματα, να μαθαίνουν από τα δεδομένα που συλλέγονται και έτσι να λαμβάνουν αποφάσεις (ή προβλέψεις), όπως η αναγνώριση φωνής [104], η πρόβλεψη [105] και η ταξινόμηση εικόνων [106]. Η βιβλιογραφία χωρίζει τις τεχνικές μηχανικής μάθησης σε δύο υποπεδία σχετικά με τη φύση του αποτελέσματος της υποκείμενης τεχνικής: (i) οι εποπτευόμενες τεχνικές μηχανικής μάθησης (supervised), οι οποίες δημιουργούν ένα μοντέλο πάνω σε ένα σύνολο δεδομένων με ετικέτες, και (ii) οι μη-εποπτευόμενες τεχνικές μηχανικής μάθησης (unsupervised), οι οποίες ομαδοποιούν δεδομένα χωρίς ετικέτες σε μικρές ομάδες με βάση τις ομοιότητές τους.

Από την άλλη, οι αλγόριθμοι βαθιάς μάθησης (Deep Learning – DL) αποτελούν έναν τύπο μηχανικής μάθησης που έχει την ικανότητα να ερμηνεύει δεδομένα, όπως κάνει ο ανθρώπινος εγκέφαλος, και να μπορεί να μαθαίνει και να ταξινομεί αντικείμενα. Αξιοποιώντας τις ικανότητες της βαθιάς μάθησης, μπορούμε να προβλέψουμε μελλοντικές καταστάσεις και να λάβουμε αποφάσεις με βάση τις τρέχουσες διαθέσιμες πληροφορίες, οι οποίες χρησιμοποιούνται ως δεδομένα εκπαίδευσης όταν εκπαιδεύουμε ένα μοντέλο DL. Αφού ολοκληρωθεί η διαδικασία της εκπαίδευσης, παράγεται ένα μοντέλο πρόβλεψης, βάσει του οποίου μπορούν να πραγματοποιηθούν προβλέψεις με βάση κάποια δεδομένα ως είσοδο. Η μηχανική μάθηση ως μια υπηρεσία (Machine Learning as a Service – MLaaS) αποτελεί μια υπηρεσία, η οποία συνήθως εκτελείται σε μια πλατφόρμα νέφους, με σκοπό να παρέχει υπηρεσίες πρόβλεψης σε διάφορους χρήστες χρησιμοποιώντας τεχνικές μηχανικής μάθησης [107]. Μια τέτοια υπηρεσία MLaaS εκτελείται σε περιβάλλον νέφους, έτσι ώστε οι χρήστες να μην χρειάζεται να δημιουργήσουν το δικό τους μοντέλο μηχανικής εκμάθησης για να κάνουν μια πρόβλεψη [108]. Ωστόσο, σε ένα τέτοιο περιβάλλον αναπτύσσονται συχνά προβλήματα σχετικά με την ασφάλεια και την ιδιωτικότητα των παρεχόμενων υπηρεσιών. Για παράδειγμα, για την εκτέλεση μιας πρόβλεψης, ο κάτοχος του μοντέλου χρειάζεται να λάβει ως είσοδο κάποια δεδομένα από τους χρήστες, ωστόσο, τα δεδομένα αυτά μπορεί να αποτελούνται από εναίσθητες πληροφορίες. Έτσι, οι χρήστες είναι ανήσυχοι και πολλές φορές δεν προτίθενται να παρέχουν τα δεδομένα τους σε μια τέτοια τρίτη υπηρεσία. Από την άλλη πλευρά, ο ιδιοκτήτης του μοντέλου ανησυχεί επίσης ότι ένας αντίπαλος θα μπορούσε να εμφανιστεί ως ένας απλός χρήστης και να προσπαθήσει να κλέψει το μοντέλο του. Επιπλέον, ένα άλλο εξίσου σημαντικό ζήτημα αποτελεί η ιδιωτικότητα του ίδιου του αποτελέσματος πρόβλεψης και εάν η πρόσβαση σε αυτό θα είναι ασφαλής από μη εξουσιοδοτημένες οντότητες. Σε αυτό το σενάριο, δηλαδή μιας υπηρεσίας MLaaS, απαιτείται ως λύση ένα σχήμα μηχανικής μάθησης (ή ειδικότερα βαθιάς μάθησης) με διασφάλιση της ιδιωτικότητας (Privacy-Preserving Machine/Deep Learning – PPML/PPDL).

Γενικά, ένα σχήμα PPDL μπορεί να χωριστεί σε τρεις κύριες φάσεις [109]: τη φάση συλλογής δεδομένων, τη φάση εκπαίδευσης (training) και τη φάση εξαγωγής συμπερασμάτων (inference). Στο Σχήμα 12.10 απεικονίζεται ένα σχήμα PPDL καθ' όλες της φάσεις λειτουργίας του. Η φάση συλλογής δεδομένων έχει ως κύριο σκοπό την ασφαλή μεταφορά δεδομένων από τον κάτοχο δεδομένων στο Νέφος. Ένας κάτοχος δεδομένων κρυπτογραφεί τα δεδομένα του χρησιμοποιώντας κάποιο σχήμα διασφάλισης της ιδιωτικότητας και τα αποστέλλει στον διακομιστή του νέφους. Με αυτόν τον τρόπο, η ασφάλεια των δεδομένων είναι εγγυημένη, καθώς μόνο ο κάτοχος των δεδομένων μπορεί να δει τις πραγματικές τιμές των δεδομένων. Στην συνέχεια, κατά την φάση εκπαίδευσης, τα κρυπτογραφημένα δεδομένα χρησιμοποιούνται ως είσοδος σε έναν αλγόριθμο βα-

θιάς μάθησης. Η εκπαιδευτική διαδικασία χωρίζεται σε μάθηση πρόσθιας τροφοδότησης (feed-forward) και σε μάθηση οπισθοδιάδοσης (backpropagation). Η μάθηση πρόσθιας τροφοδότησης εκπαιδεύει το μοντέλο, ενώ η μάθηση οπισθοδιάδοσης ελαχιστοποιεί το σφάλμα του μοντέλου. Μετά την ολοκλήρωση της φάσης εκπαίδευσης, παράγεται το μοντέλο πρόβλεψης το οποίο θα χρησιμοποιηθεί αργότερα κατά την φάση εξαγωγής συμπερασμάτων. Στη φάση εξαγωγής συμπερασμάτων, οι χρήστες που θέλουν να πραγματοποιήσουν μια πρόβλεψη στέλνουν τα ερωτήματά τους στην υπηρεσία νέφους, και χρησιμοποιώντας το μοντέλο πρόβλεψης, αποκτούν το αποτέλεσμα πρόβλεψης. Μέχρι στιγμής (2021-22) έχουν αναπτυχθεί αρκετοί αλγόριθμοι βαθιάς μάθησης που ταυτόχρονα διασφαλίζουν την ιδιωτικότητα. Οι πιο δημοφιλείς αλγόριθμοι βαθιάς μάθησης που μπορούν να υποστηρίξουν τεχνολογίες διασφάλισης της ιδιωτικότητας είναι τα νευρωνικά δίκτυα DNN (Deep Neural Network), CNN (Convolutional Neural Network) και GAN (Generative Adversarial Network).



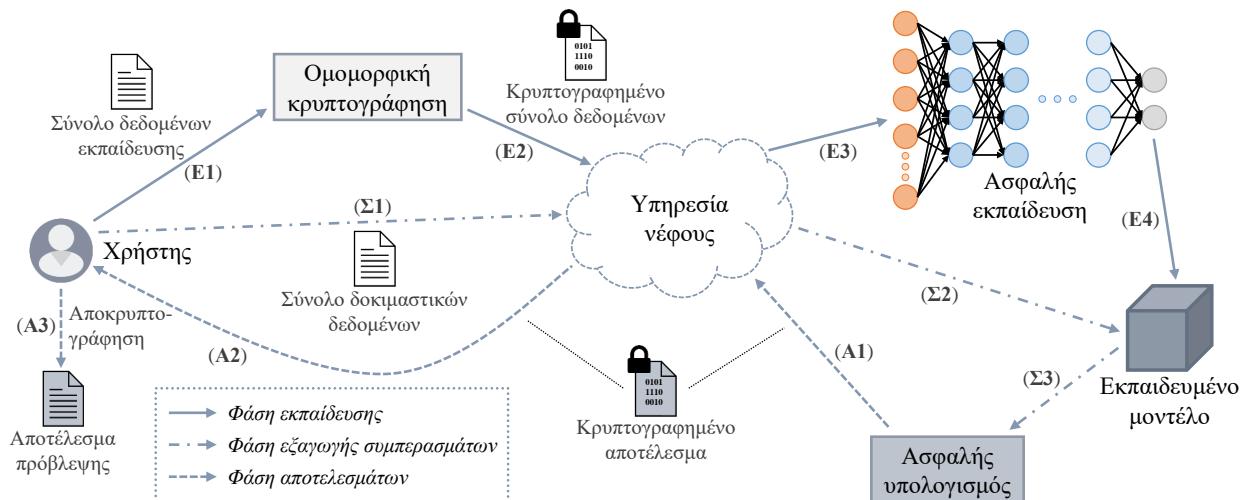
Σχήμα 12.10: Απεικόνιση ενός σχήματος βαθιάς μάθησης με διασφάλιση ιδιωτικότητας (PPDL).

Στις υποενότητες που ακολουθούν, παρουσιάζεται μια ανασκόπηση των υφιστάμενων τεχνικών κρυπτογράφησης που χρησιμοποιούνται στην μηχανική μάθηση για την διασφάλιση της ιδιωτικότητας, δίνοντας μόνο έμφαση σε λύσεις που επικεντρώνονται στην βαθιά μάθηση, και συγκεκριμένα υπηρεσιών MLaaS, μιας και ο χώρος που πραγματεύεται σε αυτήν την ενότητα είναι αρκετά μεγάλος. Αυτές οι τεχνικές ομαδοποιούνται με βάση την κύρια κρυπτογραφική τεχνική που χρησιμοποιείται και οι σημαντικότερες από αυτές είναι οι εξής: (α) ομομορφική κρυπτογράφηση, και (β) ασφαλείς υπολογισμοί πολλαπλών οντοτήτων (MPC).

12.7.1 Λύσεις PPDL με Χρήση Ομομορφικής Κρυπτογράφησης

Σε αυτή την υποενότητα, επικεντρωνόμαστε στις λύσεις PPDL [109] που βασίζονται στην ομομορφική κρυπτογράφηση. Η γενική δομή τέτοιων σχημάτων PPDL παρουσιάζεται στο Σχήμα 12.11. Όπως φαίνεται στο σχήμα, υπάρχουν τρεις κύριες φάσεις: η φάση εκπαίδευσης (E1-E2-E3-E4), η φάση εξαγωγής συμπερασμάτων (Σ1-Σ2-Σ3) και η φάση αποτελεσμάτων (A1-A2-A3). Στη φάση εκπαίδευσης, ένας χρήστης κρυπτογραφεί το σύνολο δεδομένων εκπαίδευσης χρησιμοποιώντας ομομορφική κρυπτογράφηση (E1) και στέλνει το κρυπτογραφημένο σύνολο δεδομένων στον διακομιστή νέφους (E2). Στην συνέχεια, ο διακομιστής νέφους εκτελεί μια ασφαλή εκπαίδευση (E3), η οποία έχει ως αποτέλεσμα ένα εκπαίδευμένο μοντέλο (E4). Κατά τη

φάση εξαγωγής συμπερασμάτων, αρχικά ο χρήστης στέλνει το σύνολο των δοκιμαστικών δεδομένων στον διακομιστή νέφους ($\Sigma 1$), τα οποία στη συνέχεια δίνονται ως είσοδο στο εκπαιδευμένο μοντέλο ($\Sigma 2$). Ακολουθεί η διαδικασία πρόβλεψης η οποία εκτελείται χρησιμοποιώντας το εκπαιδευμένο μοντέλο ($\Sigma 3$), με αποτέλεσμα ένα κρυπτογραφημένο αποτέλεσμα υπολογισμού. Κατά τη φάση των αποτελεσμάτων, ο διακομιστής νέφους προετοιμάζεται να μεταφέρει το κρυπτογραφημένο αποτέλεσμα υπολογισμού ($A1$) και το στέλνει στο χρήστη ($A2$). Τέλος, ο χρήστης αποκρυπτογραφεί και αποκτά το αποτέλεσμα του υπολογισμού ($A3$).



Σχήμα 12.11: Γενική δομή σχημάτων PPDL που βασίζονται σε ομομορφική κρυπτογράφηση.

Αντιπροσωπευτικές λύσεις PPDL με χρήση ομομορφικής κρυπτογράφησης: Το σχήμα *ML Confidential* αναπτύχθηκε από τον Graepel *et al.* [110] και αποτελεί ένα σχήμα PPDL που βασίζεται στους ταξινομητές Linear Means (LM) [111] και Fisher Linear Discriminant (FLD) [112], που μπορούν να υλοποιηθούν με πλήρη ομομορφική κρυπτογράφηση (RLWE, βλέπε Ενότητα 8.2.2). Το σχήμα *ML Confidential* χρησιμοποιεί μια πολυωνυμική προσέγγιση (approximation) για να υποκαταστήσει τη μη-γραμμική συνάρτηση ενεργοποίησης (activation function). Η υλοποίηση του σχήματος αυτού χρησιμοποιεί ένα σενάριο που βασίζεται σε υπηρεσίες νέφους και το κύριο χαρακτηριστικό του είναι η διασφάλιση της ιδιωτικότητας των δεδομένων του χρήστη, τόσο κατά την εκπαίδευση, όσο και κατά την εξαγωγή συμπερασμάτων. Αρχικά, ο χρήστης δημιουργεί ένα ζεύγος δημόσιου και ιδιωτικού κλειδιού, και γνωστοποιεί το δημόσιο κλειδί του στην υπηρεσία νέφους. Στη συνέχεια, ο χρήστης κρυπτογραφεί τα δεδομένα του με χρήση ομομορφικής κρυπτογράφησης και τα αποστέλλει στην υπηρεσία. Η υπηρεσία νέφους εκτελεί τη διαδικασία εκπαίδευσης χρησιμοποιώντας τα κρυπτογραφημένα δεδομένα. Στη συνέχεια, χρησιμοποιεί το παραγόμενο μοντέλο για να πραγματοποιήσει ταξινόμηση στο κρυπτογραφημένο σύνολο των δεδομένων που εισήγαγε ο χρήστης. Το αποτέλεσμα ταξινόμησης αποστέλλεται κρυπτογραφημένο στο χρήστη, ο οποίος το αποκρυπτογραφεί και μαθαίνει το αποτέλεσμα.

Αργότερα, προτάθηκε το σχήμα *CryptoNets* από τον Dowlin *et al.* [101] για την αντιμετώπιση του ζήτηματος της ιδιωτικότητας κατά την παροχή της μηχανικής μάθησης ως μια υπηρεσία (MLaaS), με βασική παραδοχή ότι υπάρχει ήδη ένα μοντέλο. Η προτεινόμενη λύση παρουσιάζει ένα πλαίσιο μηχανικής μάθησης που μπορεί να λάβει ως είσοδό του κρυπτογραφημένα δεδομένα. Το *CryptoNets* βελτιώνει αρκετά την απόδοσή του σε σχέση με το *ML Confidential* [110] χρησιμοποιώντας την πλήρη ομομορφική κρυπτογράφηση YASHE [113]. Το *CryptoNets* πραγματοποιεί προβλέψεις με βάση κρυπτογραφημένα δεδομένα και στη συνέχεια παρέχει το αποτέλεσμα πρόβλεψης, επίσης σε κρυπτογραφημένη μορφή στον χρήστη. Αργότερα, ο χρήστης μπορεί να χρησιμοποιήσει το ιδιωτικό του κλειδί για να αποκρυπτογραφήσει το αποτέλεσμα πρόβλεψης. Με αυτόν τον τρόπο, διασφαλίζεται η ιδιωτικότητα του χρήστη και η ιδιωτικότητα του αποτελέσματος πρόβλεψης. Ωστόσο, η ιδιωτικότητα του μοντέλου δεν είναι εγγυημένη, επειδή το μοντέλο υπάρχει ήδη στον διακομιστή σε μη κρυπτογραφημένη μορφή.

Το σχήμα PPDL που προτάθηκε στο [114] αποτελεί ένα μοντέλο βαθιάς μάθησης με διασφάλιση ιδιωτικότητας που βασίζεται σε μια σχετικά απλή δομή νευρωνικού δικτύου. Η προτεινόμενη λύση διορθώνει μια αδυναμία διαρροής δεδομένων των χρηστών κατά τη διάρκεια εκπαίδευσης, γνωστή ως Gradients Leak Information, που υπήρχε στο σχήμα των Shokri και Shmatikov [115]. Η βασική ιδέα στο αναθεωρημένο σχήμα PPDL [114] είναι ότι επιτρέπει στον διακομιστή νέφους να ενημερώσει το μοντέλο βαθιάς μάθησης συγκεντρώνοντας τιμές κλίσεων (gradient) από τους χρήστες. Επιπρόσθετα, οι χρήστες χρησιμοποιούν προσθετική ομομορφική κρυπτογράφηση για να προστατεύσουν τις τιμές κλίσεων τους από «αδιάκριτους» (curious) διακομιστές. Ωστόσο, και σε αυτή την προσέγγιση παραμένει μια ακόμη αδυναμία, επειδή η τρέχουσα λύση δεν αποτρέπει επιθέσεις μεταξύ των συμμετεχόντων. Για την αποφυγή αυτής της ευπάθειας, απαιτείται κατάλληλος έλεγχος ταυτότητας των συμμετεχόντων από την υπηρεσία νέφους. Αυτή η μέθοδος είναι σε θέση να αποτρέψει τη διαρροή δεδομένων κρυπτογραφώντας την τιμή κλίσης. Ωστόσο, έχει ορισμένους περιορισμούς καθώς η ομομορφική κρυπτογράφηση είναι συμβατή μόνο με τη χρήση διακομιστών για την συνάθροιση των παραμέτρων.

Επιπρόσθετα, το σχήμα TAPAS [116] προσπαθεί να αντιμετωπίσει την αδυναμία της πλήρους ομομορφικής κρυπτογράφησης, η οποία πρακτικά απαιτεί αρκετό χρόνο για την εκτέλεση μοντέλων βαθιάς μάθησης σε κρυπτογραφημένα δεδομένα [117]. Στη λύση αυτή, προτείνεται μια αρχιτεκτονική βαθιάς μάθησης που αποτελείται από ένα πλήρως συνδεδεμένο (fully-connected) επίπεδο, ένα συνελικτικό (convolutional) επίπεδο και ένα επίπεδο κανονικοποίησης παρτίδας (batch normalized) με κρυπτογραφημένους υπολογισμούς για την μείωση του χρόνου υπολογισμού. Η κύρια συμβολή αυτής της προσέγγισης είναι ένας νέος αλγόριθμος για την επιτάχυνση των δυαδικών υπολογισμών σε ένα δυαδικό νευρωνικό δίκτυο [118]. Ένα επιπλέον πλεονέκτημα του σχήματος TAPAS είναι η υποστήριξη παραλληλων υπολογιστών. Ωστόσο, ένας σημαντικός περιορισμός του είναι ότι υποστηρίζει μόνο δυαδικά νευρωνικά δίκτυα.

Ένα ακόμη ενδιαφέρον σχήμα PPDL είναι το FHE-DiNN [119] που συνδυάζει πλήρη ομομορφική κρυπτογράφηση σε ένα διακριτοποιημένο νευρωνικό δίκτυο (Discretized Neural Network – DiNN). Το σχήμα FHE-DiNN προσφέρει ένα νευρωνικό δίκτυο με γραμμική πολυπλοκότητα όσον αφορά το βάθος του δικτύου. Η γραμμικότητα αυτή επιτυγχάνεται με τη διαδικασία bootstrapping σε ένα διακριτοποιημένο νευρωνικό δίκτυο χρησιμοποιώντας έναν σταθμισμένο μέσο και μια συνάρτηση ενεργοποίησης που παίρνει τιμές μεταξύ του -1 και του 1. Ο υπολογισμός της συνάρτησης ενεργοποίησης εκτελείται κατά τη διαδικασία bootstrapping για την ανανέωση του κρυπτοκειμένου, μειώνοντας έτσι τον αθροιστικό θόρυβο που προκύπτει κατά την πλήρη ομομορφική κρυπτογράφηση. Σε σύγκριση με το CryptoNets [101], το σχήμα FHE-DiNN βελτιώνει με επιτυχία την ταχύτητα και μειώνει την πολυπλοκότητα της πλήρους ομομορφικής κρυπτογράφησης, αλλά ταυτόχρονα παρουσιάζει μείωση της ακρίβειας του μοντέλου.

Τέλος, το σχήμα E2DM [120] βασίζεται στην μετατροπή ενός συνόλου εικόνων σε πίνακες. Ο κύριος στόχος αυτής της μετατροπής είναι να μειωθεί η υπολογιστική πολυπλοκότητα, η οποία επιτυγχάνεται κρυπτογραφώντας πολλαπλούς πίνακες σε ένα μόνο κρυπτοκείμενο. Επιπρόσθετα, απαιτείται να επεκταθούν ορισμένες βασικές πράξεις πινάκων για την υποστήριξη πιο προηγμένων πράξεων, όπως ο πολλαπλασιασμός ορθογώνιων πινάκων και ο υπολογισμός του ανάστροφου ενός πίνακα. Στην προτεινόμενη λύση, όχι μόνο είναι κρυπτογραφημένα τα δεδομένα, αλλά το ίδιο το μοντέλο είναι επίσης ομομορφικά κρυπτογραφημένο. Στο σχήμα E2DM, μόνο ο χρήστης μπορεί να αποκρυπτογράφησε το αποτέλεσμα πρόβλεψης. Όσον αφορά την βαθιά μάθηση, το E2DM χρησιμοποιεί συνελικτικά νευρωνικά δίκτυα (CNN) με ένα συνελικτικό επίπεδο, δύο πλήρως συνδεδεμένα επίπεδα και μια τετραγωνική συνάρτηση ενεργοποίησης (δηλ., $\sigma_{square}(x) = x^2$). Το κύριο μειονέκτημα του σχήματος E2DM είναι ότι μπορεί να υποστηρίξει μόνο απλές πράξεις πινάκων.

12.7.2 Λύσεις PPDL με Χρήση Ασφαλών Υπολογισμών Πολλαπλών Οντοτήτων

Η γενική δομή των λύσεων PPDL που βασίζονται σε ασφαλείς υπολογισμούς πολλαπλών οντοτήτων (MPC), κατά την φάση εκπαίδευσης ενός αλγορίθμου βαθιάς μάθησης [109], παρουσιάζεται στο Σχήμα 12.12. Με βάση αυτό, τα βήματα που πραγματοποιούνται σε κάθε επανάληψη της εκπαίδευσης είναι τα εξής:

1^o Βήμα: Οι χρήστες πραγματοποιούν τοπική εκπαίδευση (local training) χρησιμοποιώντας τα προσωπικά

τους δεδομένα.

2^o Βήμα: Το αποτέλεσμα (δηλ., κλίσεις ή βάρη) από τη διαδικασία της τοπικής εκπαίδευσης μετατρέπεται σε μυστικά μερίδια.

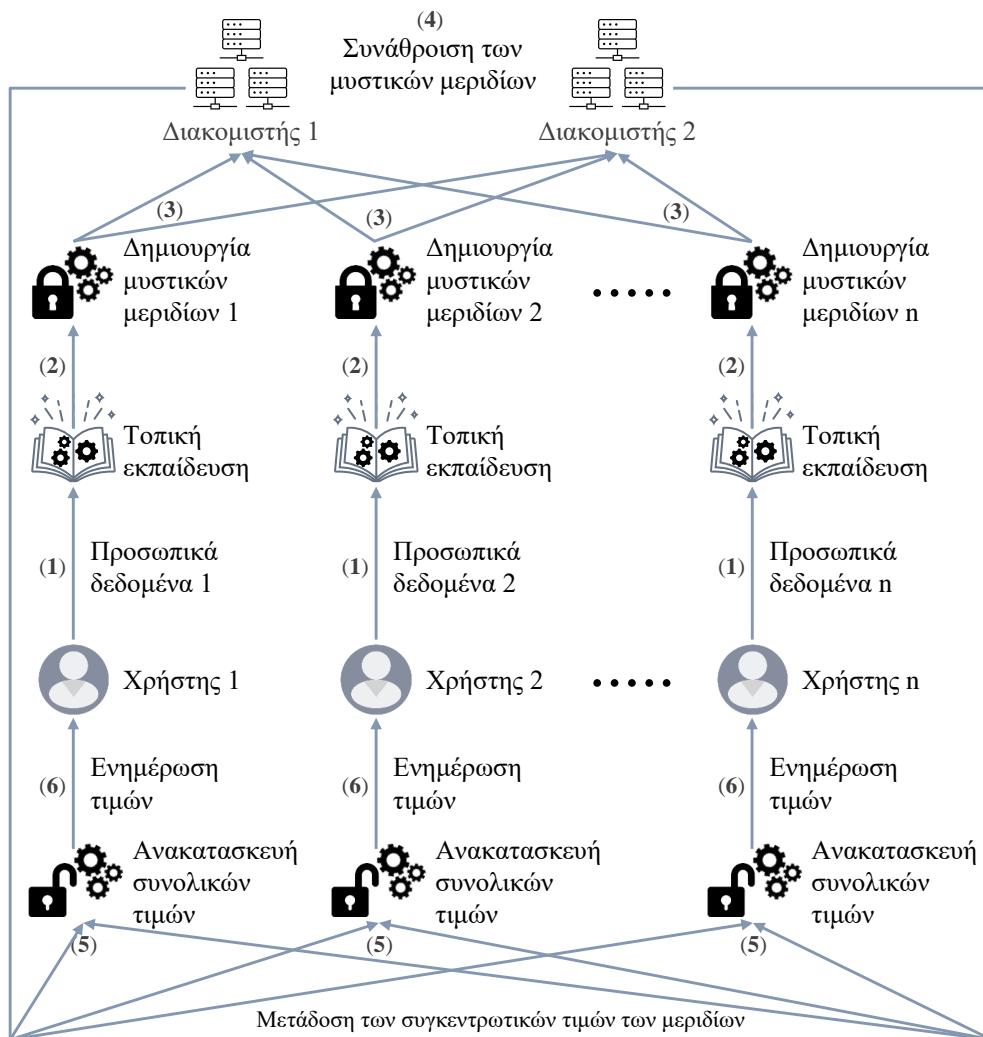
3^o Βήμα: Τα μυστικά μερίδια μεταδίδονται σε κάθε διακομιστή.

4^o Βήμα: Οι διακομιστές συναθροίζουν τα μυστικά μερίδια από τους χρήστες.

5^o Βήμα: Η συγκεντρωτική τιμή των μεριδίων μεταδίδεται από κάθε διακομιστή σε κάθε χρήστη.

6^o Βήμα: Κάθε χρήστης ανακατασκευάζει τις συνολικές τιμές από τις επιμέρους τιμές που έλαβε και ενημερώνει τις τοπικές τιμές του για την επόμενη επανάληψη της διαδικασίας εκπαίδευσης.

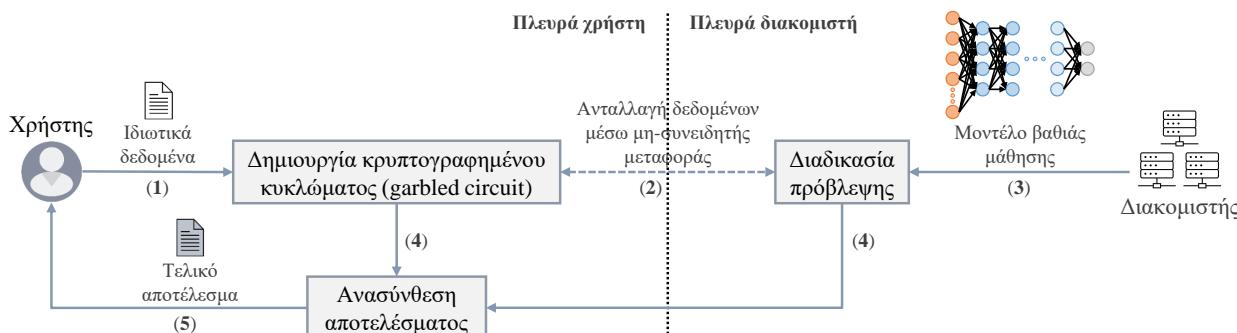
Στην περίπτωση ασφαλών υπολογισμών πολλαπλών οντοτήτων (MPC), η κοινή χρήση μυστικών (secret sharing) χρησιμοποιείται για τη διασφάλιση της ιδιωτικότητας των δεδομένων. Ωστόσο, για συγκεκριμένους ασφαλείς υπολογισμούς δύο οντοτήτων (2PC), χρησιμοποιείται συνήθως ένα Garbled Circuit (βλέπε Ενότητα 8.1.1) αντί για την κοινή χρήση μυστικών.



Σχήμα 12.12: Γενική δομή σχημάτων PPDL κατά την φάση εκπαίδευσης που βασίζονται σε ασφαλείς υπολογισμούς πολλαπλών οντοτήτων (MPC).

Αντίστοιχα, η γενική δομή των λύσεων PPDL κατά την φάση εξαγωγής συμπερασμάτων [109], όπως είναι αναμενόμενο, βασίζεται αποκλειστικά σε ασφαλείς υπολογισμούς δύο οντοτήτων (2PC) και παρουσιάζεται στο Σχήμα 12.13. Στους ασφαλείς υπολογισμούς δύο οντοτήτων, ο χρήστης χρησιμοποιεί ένα Garbled Circuit (βλέπε Ενότητα 8.1.1) για να προστατεύσει την ιδιωτικότητα των δεδομένων του. Η επικοινωνία μεταξύ του χρήστη και του διακομιστή είναι ασφαλής κάνοντας χρήση του πρωτοκόλλου της μη-συνειδητής μεταφοράς (Oblivious Transfer – OT) (βλέπε Ορισμό 8.2 στην Ενότητα 8.1.1). Με βάση το Σχήμα 12.13, τα βήματα που πραγματοποιούνται κατά την φάση εξαγωγής συμπερασμάτων είναι τα εξής:

- 1^o Βήμα:** Ο χρήστης κάνοντας χρήση ενός προσυμφωνημένου λογικού δυαδικού κυκλώματος και της ιδιωτικής του εισόδου παράγει το κρυπτογραφημένο κύκλωμα (garbled circuit).
- 2^o Βήμα:** Πραγματοποιείται ανταλλαγή δεδομένων μεταξύ του χρήστη και του διακομιστή κάνοντας χρήση του πρωτοκόλλου της μη-συνειδητής μεταφοράς.
- 3^o Βήμα:** Αφού ολοκληρωθεί η ανταλλαγή δεδομένων, ο διακομιστής πραγματοποιεί τη διαδικασία πρόβλεψης, χρησιμοποιώντας τα δεδομένα ως είσοδο στο μοντέλο βαθιάς μάθησης.
- 4^o Βήμα:** Το αποτέλεσμα της πρόβλεψης αποστέλλεται πίσω στο χρήστη και αυτός με την σειρά του, γνωρίζοντας τον πίνακα αντικαταστάσεων κατά την κρυπτογράφηση του κυκλώματος, ανασυνθέτει το αποτέλεσμα.
- 5^o Βήμα:** Ο χρήστης λαμβάνει το τελικό αποτέλεσμα.



Σχήμα 12.13: Γενική δομή σχημάτων PPDL κατά την φάση εξαγωγής συμπερασμάτων που βασίζονται σε ασφαλείς υπολογισμούς δύο οντοτήτων (2PC).

Αντιπροσωπευτικές λύσεις PPDL με χρήση ασφαλών υπολογισμών πολλαπλών οντοτήτων: Το σχήμα *SecureML* [121] αποτελεί ένα νέο πρωτόκολλο PPML που χρησιμοποιεί πρωτόκολλα μη-συνειδητής μεταφοράς, Garbled Circuit, και κοινής χρήσης μυστικών για την διασφάλιση της ιδιωτικότητας κατά τη φάση εκπαίδευσης ενός νευρωνικού δικτύου. Όσον αφορά την βαθιά μάθηση, έχει την δυνατότητα να πραγματοποιεί γραμμική και λογιστική παλινδρόμηση σε ένα περιβάλλον νευρωνικών δικτύων DNN (Deep Neural Network). Το σχήμα *SecureML* προτείνει έναν αλγόριθμο πρόσθεσης και πολλαπλασιασμού για τις τιμές της κοινής χρήσης μυστικών στη γραμμική παλινδρόμηση. Επιπλέον, η μέθοδος Stochastic Gradient Descent (SGD) χρησιμοποιείται για τον υπολογισμό της βέλτιστης τιμής παλινδρόμησης. Το κύριο μειονέκτημα αυτού του σχήματος είναι ότι μπορεί να εφαρμοστεί μόνο ένα απλό νευρωνικό δίκτυο, χωρίς κανένα συνελικτικό επίπεδο, με αποτέλεσμα η ακρίβεια που παρέχει να είναι αρκετά χαμηλή. Μια επιπλέον αδυναμία του είναι ότι βασίζεται στην υπόθεση ότι οι διακομιστές δεν συνεννοούνται (non-colluding) μεταξύ τους, καθώς μπορεί να είναι μη-έμπιστοι. Σε αντίθεση περίπτωση, η ιδιωτικότητα των συμμετεχόντων χρηστών δεν μπορεί να εγγυηθεί.

Από την άλλη, το σχήμα *MiniONN* [122] αποτελεί ένα πλαίσιο διασφάλισης ιδιωτικότητας κατά τη φάση εξαγωγής συμπερασμάτων (προβλέψεων) που βασίζεται στον μετασχηματισμό ενός υπάρχοντος νευρωνικού δίκτυου σε ένα μη-συνειδητό νευρωνικό δίκτυο (*Oblivious Neural Network – ONN*). Η διαδικασία μετασχηματισμού στο *MiniONN* περιλαμβάνει μη-γραμμικές συναρτήσεις, έχοντας ως τίμημα μια μικρή σχετικά απώλεια ακρίβειας. Υπάρχουν δύο είδη μετασχηματισμών που παρέχονται από το σχήμα *MiniONN*, συμπεριλαμβανομένου του μη-συνειδητού μετασχηματισμού για τη τμηματική γραμμική (piecewise linear) συνάρτηση ενεργοποίησης και του μη-συνειδητού μετασχηματισμού για την ομαλή (smooth) συνάρτηση ενεργοποίησης. Μια ομαλή συνάρτηση ενεργοποίησης μπορεί να μετασχηματιστεί σε ένα συνεχές πολυώνυμο χωρίζοντας τη συνάρτηση σε πολλά τμήματα. Στη συνέχεια, για κάθε τμήμα χρησιμοποιείται μια πολυωνυμική προσέγγιση (approximation) για να υποκαταστήσει αυτό το τμήμα της ομαλής συνάρτησης, με αποτέλεσμα την δημιουργία μιας τμηματικής γραμμικής συνάρτησης. Ως εκ τούτου, το σχήμα *MiniONN* μπορεί να υποστηρίξει όλες τις συναρτήσεις ενεργοποίησης που έχουν μονοτονικό εύρος, τμηματικά πολυώνυμο, ή μπορούν να προσεγγιστούν με πολυωνυμικές συναρτήσεις. Η πειραματική διαδικασία έδειξε ότι το σχήμα *MiniONN* ξεπερνά το σχήμα *CryptoNets* [101] και το σχήμα *SecureML* [121] όσον αφορά το μέγεθος και την καθυστέρηση των μηνυμάτων. Η κύρια αδυναμία του είναι ότι δεν υποστηρίζει την επεξεργασία σε παρτίδες (batches). Επιπλέον, το *MiniONN* βασίζεται στο μοντέλο των αδιάκριτων αλλά ειλικρινών (*honest-but-curious*) αντιπάλων, επομένως δεν μπορεί να παρέχει ασφάλεια έναντι κακόβουλων αντιπάλων.

Το σχήμα *ABY³* που προτάθηκε από τον Mohassel *et al.* [123], αποτελεί ένα πρωτόκολλο PPML που βασίζεται σε υπολογισμούς τριών οντοτήτων (3PC) για την εκπαίδευση μοντέλων. Η κύρια συνεισφορά αυτού του πρωτοκόλλου είναι η ικανότητά του να επιλέγει το κατάλληλο σχήμα κοινής χρήσης μυστικού (αριθμητικό, δυαδικό και Garbled Circuit του Yao) ανάλογα με τις ανάγκες επεξεργασίας. Ο κύριος σκοπός του *ABY³* είναι να επιλύσει ένα κλασικό πρόβλημα στο PPDL που χρειάζεται εναλλαγή μεταξύ αριθμητικών (για παράδειγμα πρόσθεσης και πολλαπλασιασμού) και μη αριθμητικών πράξεων (όπως προσεγγίσεις των συναρτήσεων ενεργοποίησης). Το σχήμα *ABY³* μπορεί να χρησιμοποιηθεί για την εκπαίδευση μοντέλων γραμμικής παλινδρόμησης, λογιστικής παλινδρόμησης και νευρωνικών δίκτυων. Το αριθμητικό σχήμα κοινής χρήσης μυστικού χρησιμοποιείται κατά την εκπαίδευση μοντέλων γραμμικής παλινδρόμησης, ενώ για τον υπολογισμό της λογιστικής παλινδρόμησης και των μοντέλων νευρωνικών δίκτυων, χρησιμοποιείται η δυαδική κοινή χρήση μυστικού σε Garbled Circuits τριών οντοτήτων. Το πλαίσιο που παρέχει το σχήμα *ABY³* είναι ασφαλές έναντι κακόβουλων αντιπάλων, οπότε παρουσιάζει μεγαλύτερο επίπεδο ασφάλειας από τα προαναφερθέντα σχήματα.

Το σχήμα *Chameleon* [124] αποτελεί μια PPDL μέθοδο που συνδυάζει υπολογισμούς MPC και συνελικτικά νευρωνικά δίκτυα (CNN) για την ασφαλή εξαγωγή συμπερασμάτων (προβλέψεων). Όσον αφορά την ιδιωτικότητα, το σχήμα *Chameleon* χρησιμοποιεί το Garbled Circuit του Yao, το οποίο επιτρέπει σε δύο οντότητες να πραγματοποιούν κοινούς υπολογισμούς χωρίς να αποκαλύπτουν τις εισόδους δεδομένων τους. Το προτεινόμενο σχήμα διαθέτει δύο φάσεις: μια διαδικτυακή φάση και μια φάση εκτός σύνδεσης. Κατά τη διάρκεια της διαδικτυακής φάσης επιτρέπεται σε όλα τα μέρη να επικοινωνούν, ενώ κατά τη φάση εκτός σύνδεσης προϋπολογίζονται οι κρυπτογραφικές πράξεις. Το σχήμα *Chameleon* επιτυγχάνει ταχύτερη εκτέλεση σε σύγκριση με τα σχήματα *CryptoNets* [101] και *MiniONN* [122]. Το σχήμα *Chameleon* απαιτεί δύο διακομιστές που δεν συνεννοούνται (non-colluding) μεταξύ τους για να διασφαλίσει την ιδιωτικότητα και την ασφάλεια των δεδομένων. Κατά τη φάση εξαγωγής ιδιωτικών συμπερασμάτων, απαιτείται μια ανεξάρτητη τρίτη οντότητα (ή ένα ασφαλές υλικό όπως το Intel SGX). Το μοντέλο ασφαλείας του σχήματος *Chameleon* βασίζεται σε αδιάκριτους αλλά ειλικρινείς (*honest-but-curious*) αντιπάλους, χωρίς να μπορεί να χειριστεί βέβαια κακόβουλους αντιπάλους. Το πρωτόκολλο του *Chameleon* βασίζεται σε υπολογισμούς δύο οντοτήτων (2PC), επομένως δεν είναι αποτελεσματικό να εφαρμοστεί σε περισσότερες από δύο οντότητες.

Τέλος, το σχήμα *SecureNN* [125] παρέχει το πρώτο σύστημα που διασφαλίζει την ιδιωτικότητα τόσο κατά την εκπαίδευση όσο και κατά την εξαγωγή συμπερασμάτων σε πολύπλοκα νευρωνικά δίκτυα, και μάλιστα παρέχοντας ασφάλεια έναντι αδιάκριτων αλλά ειλικρινών (*honest-but-curious*) αντιπάλων καθώς και κακόβουλων αντιπάλων. Το σύστημα βασίζεται σε πρωτόκολλα MPC και σε συνελικτικά νευρωνικά δίκτυα (CNN). Το σχήμα *SecureNN* εμφανίζεται να είναι 2–4 φορές ταχύτερο από άλλα σχήματα PPDL που βασίζονται σε

MPC, όπως τα SecureML [121], MiniONN [122] και Chameleon [124]. Η κύρια συνεισφορά της προτεινόμενης λύσης είναι η ανάπτυξη ενός νέου πρωτοκόλλου για λογικούς (boolean) υπολογισμούς (όπως στις ReLU, Maxpool, αλλά και σε παραλλαγές τους) που διασφαλίζει μικρότερο κόστος επικοινωνίας από ότι το Garbled Circuit του Yao. Αυτός είναι ο λόγος για τον οποίο το σχήμα SecureNN επιτυγχάνει ταχύτερο χρόνο εκτέλεσης από τα άλλα σχήματα που αναφέρθηκαν προηγούμενως. Η κύρια αδυναμία του SecureNN εντοπίζεται στο ότι έχει μεγαλύτερο κόστος επικοινωνίας σε σύγκριση με το ABY³ [123]. Ωστόσο, εάν το πρωτόκολλο SecureNN τροποποιηθεί έτσι ώστε να χρησιμοποιεί πολλαπλασιασμό πινάκων, όπως το ABY³, ο αριθμός των γύρων επικοινωνίας θα μειωθεί αρκετά.

Βιβλιογραφία

- [1] European Parliament and Council. “Regulation (EU) 2016/679 of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)”. In: *Official Journal of the European Union* (2016), pp. 1–88.
- [2] Michael Barbaro and Tom Zeller. *A Face is Exposed for AOL Searcher No. 4417749*. News Media, accessed on 29 August 2022. <http://www.nytimes.com/2006/08/09/technology/09aol.html>. Aug. 2006.
- [3] David L. Chaum. “Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms”. In: *Communications of the ACM* 24.2 (Feb. 1981), pp. 84–90. ISSN: 0001-0782. doi: 10.1145/358549.358563.
- [4] Oliver Berthold, Hannes Federrath, and Stefan Köpsell. “Web MIXes: A System for Anonymous and Unobservable Internet Access”. In: *Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability*. Ed. by Hannes Federrath. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 115–129. ISBN: 978-3-540-44702-3. doi: 10.1007/3-540-44702-4_7.
- [5] Roger Dingledine, Nick Mathewson, and Paul Syverson. “TOR: The Second-Generation Onion Router”. In: *Proceedings of the 13th USENIX Security Symposium (USENIX Security '04)*. San Diego, CA: USENIX Association, Aug. 2004.
- [6] The Invisible Internet Project. *I2P Anonymous Network*. Official Website, accessed on 29 August 2022. <http://www.i2p2.de>. 2022.
- [7] Michael K. Reiter and Aviel D. Rubin. “Anonymous Web Transactions with Crowds”. In: *Communications of the ACM* 42.2 (Feb. 1999), pp. 32–48. ISSN: 0001-0782. doi: 10.1145/293411.293778.
- [8] Michael J. Freedman and Robert Morris. “Tarzan: A Peer-to-Peer Anonymizing Network Layer”. In: *Proceedings of the 9th ACM Conference on Computer and Communications Security*. CCS '02. Washington, DC, USA: Association for Computing Machinery, 2002, pp. 193–206. ISBN: 15-8113-612-9. doi: 10.1145/586110.586137.
- [9] David Goldschlag, Michael Reed, and Paul Syverson. “Onion Routing”. In: *Communications of the ACM* 42.2 (Feb. 1999), pp. 39–41. ISSN: 0001-0782. doi: 10.1145/293411.293443.
- [10] Aggelos Kiayias and Moti Yung. “Self-Tallying Elections and Perfect Ballot Secrecy”. In: *Public Key Cryptography*. Ed. by David Naccache and Pascal Paillier. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 141–158. ISBN: 978-3-540-45664-3. doi: 10.1007/3-540-45664-3_10.
- [11] Kun Peng. “An Efficient Shuffling Based eVoting Scheme”. In: *Journal of Systems and Software* 84.6 (2011), pp. 906–922. ISSN: 0164-1212. doi: 10.1016/j.jss.2011.01.001.

- [12] Htet Ne Oo and AM Aung. "A Survey of Different Electronic Voting Systems". In: *International Journal of Scientific Engineering and Technology Research* 3.16 (2014), pp. 3460–3464.
- [13] Francesc Sebé, Josep M Miret, Jordi Pujolàs, and Jordi Puiggalí. "Simple and Efficient Hash-Based Verifiable Mixing for Remote Electronic Voting". In: *Computer Communications* 33.6 (2010), pp. 667–675. ISSN: 0140-3664. DOI: 10.1016/j.comcom.2009.11.013.
- [14] Markus Jakobsson, Ari Juels, and Ronald L. Rivest. "Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking". In: *Proceedings of the 11th USENIX Security Symposium (USENIX Security '02)*. San Francisco, CA: USENIX Association, Aug. 2002.
- [15] Yun-Xing Kho, Swee-Huay Heng, and Ji-Jian Chin. "A Review of Cryptographic Electronic Voting". In: *Symmetry* 14.5 (2022). ISSN: 2073-8994. DOI: 10.3390/sym14050858.
- [16] Ronald L Rivest, Adi Shamir, and Leonard Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Communications of the ACM* 21.2 (Feb. 1978), pp. 120–126. ISSN: 0001-0782. DOI: 10.1145/359340.359342.
- [17] Choonsik Park, Kazutomo Itoh, and Kaoru Kurosawa. "Efficient Anonymous Channel and All/Nothing Election Scheme". In: *Advances in Cryptology — EUROCRYPT '93*. Ed. by Tor Helle-seth. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 248–259. ISBN: 978-3-540-48285-7. DOI: 10.1007/3-540-48285-7_21.
- [18] Taher Elgamal. "A public key cryptosystem and a signature scheme based on discrete logarithms". In: *IEEE Transactions on Information Theory* 31.4 (1985), pp. 469–472. DOI: 10.1109/TIT.1985.1057074.
- [19] Pascal Paillier. "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes". In: *Advances in Cryptology — EUROCRYPT '99*. Ed. by Jacques Stern. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 223–238. ISBN: 978-3-540-48910-8.
- [20] Kun Peng and Feng Bao. "Efficient Multiplicative Homomorphic E-Voting". In: *Information Security*. Ed. by Mike Burmester, Gene Tsudik, Spyros Magliveras, and Ivana Ilić. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 381–393. ISBN: 978-3-642-18178-8. DOI: 10.1007/978-3-642-18178-8_32.
- [21] Craig Gentry. "Fully Homomorphic Encryption Using Ideal Lattices". In: *Forty-First Annual ACM Symposium on Theory of Computing*. STOC '09. Bethesda, MD, USA: ACM, 2009, pp. 169–178. ISBN: 9781605585062. DOI: 10.1145/1536414.1536440.
- [22] Josh D. Cohen and Michael J. Fischer. "A Robust and Verifiable Cryptographically Secure Election Scheme". In: *Proceedings of the 26th Annual Symposium on Foundations of Computer Science (SFCS 1985)*. Portland, OR, USA: IEEE, 1985, pp. 372–382. DOI: 10.1109/SFCS.1985.2.
- [23] Kun Peng, Riza Aditya, Colin Boyd, Ed Dawson, and Byoungcheon Lee. "Multiplicative Homomorphic E-Voting". In: *Progress in Cryptology - INDOCRYPT 2004*. Ed. by Anne Canteaut and Kapaleeswaran Viswanathan. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 61–72. ISBN: 978-3-540-30556-9. DOI: 10.1007/978-3-540-30556-9_6.
- [24] David Chaum. "Blind Signatures for Untraceable Payments". In: *Advances in Cryptology*. Ed. by David Chaum, Ronald L. Rivest, and Alan T. Sherman. Boston, MA: Springer US, 1983, pp. 199–203. ISBN: 978-1-4757-0602-4. DOI: 10.1007/978-1-4757-0602-4_18.
- [25] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. "A Practical Secret Voting Scheme for Large Scale Elections". In: *Advances in Cryptology — AUSCRYPT '92*. Ed. by Jennifer Seberry and Yu-liang Zheng. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 244–251. ISBN: 978-3-540-47976-5. DOI: 10.1007/3-540-57220-1_66.

- [26] Fangguo Zhang and Kwangjo Kim. "ID-Based Blind Signature and Ring Signature from Pairings". In: *Advances in Cryptology — ASIACRYPT 2002*. Ed. by Yuliang Zheng. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 533–547. ISBN: 978-3-540-36178-7. doi: 10.1007/3-540-36178-2_33.
- [27] Mahender Kumar, Satish Chand, and C. P. Katti. "A Secure End-to-End Verifiable Internet-Voting System Using Identity-Based Blind Signature". In: *IEEE Systems Journal* 14.2 (2020), pp. 2032–2041. doi: 10.1109/JST.2019.2940474.
- [28] Freya Sheer Hardwick, Apostolos Gioulis, Raja Naeem Akram, and Konstantinos Markantonakis. "E-Voting With Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy". In: *IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 2018, pp. 1561–1567. doi: 10.1109/Cybermatics_2018.2018.00262.
- [29] Uzma Jafar and Mohd Juzaiddin Ab Aziz. "A State of the Art Survey and Research Directions on Blockchain Based Electronic Voting System". In: *Advances in Cyber Security*. Ed. by Mohammed Anbar, Nibras Abdullah, and Selvakumar Manickam. Singapore: Springer Singapore, 2021, pp. 248–266. ISBN: 978-981-33-6835-4. doi: 10.1007/978-981-33-6835-4_17.
- [30] Yi Liu and Qi Wang. *An E-voting Protocol Based on Blockchain*. Cryptology ePrint Archive, Paper 2017/1043. <https://ia.cr/2017/1043>. 2017.
- [31] Sunoo Park, Michael Specter, Neha Narula, and Ronald L Rivest. "Going from Bad to Worse: From Internet Voting to Blockchain Voting". In: *Journal of Cybersecurity* 7.1 (Feb. 2021). ISSN: 2057-2085. doi: 10.1093/cybsec/tyaa025.
- [32] National Academies of Sciences, Engineering, and Medicine. *Securing the Vote: Protecting American Democracy*. Washington, DC: National Academies Press, 2018. doi: 10.17226/25120.
- [33] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. "A Homomorphic LWE Based E-voting Scheme". In: *Post-Quantum Cryptography*. Ed. by Tsuyoshi Takagi. Cham: Springer International Publishing, 2016, pp. 245–265. ISBN: 978-3-319-29360-8. doi: 10.1007/978-3-319-29360-8_16.
- [34] Guillaume Kaim, Sébastien Canard, Adeline Roux-Langlois, and Jacques Traoré. "Post-quantum Online Voting Scheme". In: *Financial Cryptography and Data Security. FC 2021 International Workshops*. Ed. by Matthew Bernhard et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2021, pp. 290–305. ISBN: 978-3-662-63958-0. doi: 10.1007/978-3-662-63958-0_25.
- [35] Hua Dong and Li Yang. *A Voting Scheme with Post-Quantum Security Based on Physical Laws*. Cryptology ePrint Archive, Paper 2018/446. <https://ia.cr/2018/446>. 2018.
- [36] Rui Xu, Liusheng Huang, Wei Yang, and Libao He. "Quantum group blind signature scheme without entanglement". In: *Optics Communications* 284.14 (2011), pp. 3654–3658. ISSN: 0030-4018. doi: 10.1016/j.optcom.2011.03.083.
- [37] Ramiro Alvarez and Mehrdad Nojoumian. "Comprehensive Survey on Privacy-Preserving Protocols for Sealed-Bid Auctions". In: *Computers & Security* 88 (2020), p. 101502. ISSN: 0167-4048. doi: 10.1016/j.cose.2019.03.023.
- [38] Hiroaki Kikuchi, Michael Hakavy, and Doug Tygar. "Multi-Round Anonymous Auction Protocols". English. In: *IEICE Transactions on Information and Systems* E82-D.4 (1999), pp. 769–777. ISSN: 0916-8532.

- [39] Christian Cachin. "Efficient Private Bidding and Auctions with an Oblivious Third Party". In: *Proceedings of the 6th ACM Conference on Computer and Communications Security*. CCS '99. Kent Ridge Digital Labs, Singapore: Association for Computing Machinery, 1999, pp. 120–127. ISBN: 15-8113-148-8. DOI: 10.1145/319709.319726.
- [40] Kouichi Sakurai and Shingo Miyazaki. "An Anonymous Electronic Bidding Protocol Based on a New Convertible Group Signature Scheme". In: *Information Security and Privacy*. Ed. by E. P. Dawson, A. Clark, and Colin Boyd. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 385–399. ISBN: 978-3-540-45030-6. DOI: 10.1007/10718964_32.
- [41] Kazue Sako. "An Auction Protocol Which Hides Bids of Losers". In: *Public Key Cryptography*. Ed. by Hideki Imai and Yuliang Zheng. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 422–432. ISBN: 978-3-540-46588-1. DOI: 10.1007/978-3-540-46588-1_28.
- [42] Koutarou Suzuki, Kunio Kobayashi, and Hikaru Morita. "Efficient Sealed-bid Auction using Hash Chain". In: *Information Security and Cryptology — ICISC 2000*. Ed. by Dongho Won. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 183–191. ISBN: 978-3-540-45247-8. DOI: 10.1007/3-540-45247-8_15.
- [43] Matthew K. Franklin and Michael K. Reiter. "The Design and Implementation of a Secure Auction Service". In: *IEEE Transactions on Software Engineering* 22.5 (1996), pp. 302–312. DOI: 10.1109/32.502223.
- [44] Khanh Quoc Nguyen and Jacques Traoré. "An Online Public Auction Protocol Protecting Bidder Privacy". In: *Information Security and Privacy*. Ed. by E. P. Dawson, A. Clark, and Colin Boyd. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 427–442. ISBN: 978-3-540-45030-6. DOI: 10.1007/10718964_35.
- [45] Mehrdad Nojoumian and Douglas R. Stinson. "Unconditionally Secure First-Price Auction Protocols Using a Multicomponent Commitment Scheme". In: *Information and Communications Security*. Ed. by Miguel Soriano, Sihan Qing, and Javier López. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 266–280. ISBN: 978-3-642-17650-0. DOI: 10.1007/978-3-642-17650-0_19.
- [46] Hannu Nurmi and Arto Salomaa. "Cryptographic protocols for Vickrey auctions". In: *Group Decision and Negotiation* 2.4 (1993), pp. 363–373. DOI: 10.1007/BF01384489.
- [47] Moni Naor, Benny Pinkas, and Reuban Sumner. "Privacy Preserving Auctions and Mechanism Design". In: *Proceedings of the 1st ACM Conference on Electronic Commerce*. EC '99. Denver, Colorado, USA: Association for Computing Machinery, 1999, pp. 129–139. ISBN: 1581131763. DOI: 10.1145/336992.337028.
- [48] Helger Lipmaa, N. Asokan, and Valtteri Niemi. "Secure Vickrey Auctions without Threshold Trust". In: *Financial Cryptography*. Ed. by Matt Blaze. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 87–101. ISBN: 978-3-540-36504-4. DOI: 10.1007/3-540-36504-4_7.
- [49] Ari Juels and Michael Szydlo. "A Two-Server, Sealed-Bid Auction Protocol". In: *Financial Cryptography*. Ed. by Matt Blaze. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 72–86. ISBN: 978-3-540-36504-4. DOI: 10.1007/3-540-36504-4_6.
- [50] Mehrdad Nojoumian and Douglas R. Stinson. "Efficient Sealed-Bid Auction Protocols Using Verifiable Secret Sharing". In: *Information Security Practice and Experience*. Ed. by Xinyi Huang and Jianying Zhou. Cham: Springer International Publishing, 2014, pp. 302–317. ISBN: 978-3-319-06320-1. DOI: 10.1007/978-3-319-06320-1_23.

- [51] Douglas R. Stinson and Ruizhong Wei. "Unconditionally Secure Proactive Secret Sharing Scheme with Combinatorial Structures". In: *Selected Areas in Cryptography*. Ed. by Howard Heys and Carlisle Adams. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 200–214. ISBN: 978-3-540-46513-3. DOI: [10.1007/3-540-46513-8_15](https://doi.org/10.1007/3-540-46513-8_15).
- [52] Tal Rabin and Michael Ben-Or. "Verifiable Secret Sharing and Multiparty Protocols with Honest Majority". In: *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*. STOC '89. Seattle, Washington, USA: Association for Computing Machinery, 1989, pp. 73–85. ISBN: 0897913078. DOI: [10.1145/73007.73014](https://doi.org/10.1145/73007.73014).
- [53] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. "Private Information Retrieval". In: *Proceedings of the 36th Annual Symposium on the Foundations of Computer Science*. FOCS '95. Washington, DC, USA: IEEE Computer Society, 1995, pp. 41–50. ISBN: 0818671831.
- [54] Femi George Olumofin. "Practical Private Information Retrieval". <http://hdl.handle.net/10012/6142>. PhD thesis. Waterloo, Ontario, Canada, 2011.
- [55] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. "Private Information Retrieval". In: *Journal of the ACM* 45.6 (Nov. 1998), pp. 965–981. ISSN: 0004-5411. DOI: [10.1145/293347.293350](https://doi.org/10.1145/293347.293350).
- [56] Jan Camenisch, Maria Dubovitskaya, and Gregory Neven. "Unlinkable Priced Oblivious Transfer with Rechargeable Wallets". In: *Financial Cryptography and Data Security*. Ed. by Radu Sion. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 66–81. ISBN: 978-3-642-14577-3. DOI: [10.1007/978-3-642-14577-3_8](https://doi.org/10.1007/978-3-642-14577-3_8).
- [57] ICANN Security and Stability Advisory Committee (SSAC). *Report on Domain Name Front Running*. Report - SAC 024, accessed on 31 August 2022. <https://www.icann.org/en/system/files/files/sac-024-en.pdf>. Feb. 2008.
- [58] Mikhail Bilenko, Matthew Richardson, and Janice Tsai. "Targeted, Not Tracked: Client-Side Solutions for Privacy-Friendly Behavioral Advertising". In: *Proceedings of the 4th Hot Topics in Privacy Enhancing Technologies*. HotPETs '11. 2011, pp. 29–40.
- [59] Eyal Kushilevitz and Rafail Ostrovsky. "Replication is Not Needed: Single Database, Computationally-Private Information Retrieval". In: *Proceedings 38th Annual Symposium on Foundations of Computer Science*. IEEE, 1997, pp. 364–373. DOI: [10.1109/SFCS.1997.646125](https://doi.org/10.1109/SFCS.1997.646125).
- [60] Julien P. Stern. "A New and Efficient All-Or-Nothing Disclosure of Secrets Protocol". In: *Advances in Cryptology — ASIACRYPT'98*. Ed. by Kazuo Ohta and Dingyi Pei. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 357–371. ISBN: 978-3-540-49649-6. DOI: [10.1007/3-540-49649-1_28](https://doi.org/10.1007/3-540-49649-1_28).
- [61] Helger Lipmaa. "An Oblivious Transfer Protocol with Log-Squared Communication". In: *Information Security*. Ed. by Jianying Zhou, Javier Lopez, Robert H. Deng, and Feng Bao. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 314–328. ISBN: 978-3-540-31930-6. DOI: [10.1007/11556992_23](https://doi.org/10.1007/11556992_23).
- [62] Christian Cachin, Silvio Micali, and Markus Stadler. "Computationally Private Information Retrieval with Polylogarithmic Communication". In: *Advances in Cryptology — EUROCRYPT '99*. Ed. by Jacques Stern. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 402–414. ISBN: 978-3-540-48910-8. DOI: [10.1007/3-540-48910-X_28](https://doi.org/10.1007/3-540-48910-X_28).
- [63] Carlos Aguilar Melchor and Philippe Gaborit. *A Lattice-Based Computationally-Efficient Private Information Retrieval Protocol*. Cryptology ePrint Archive, Paper 2007/446. <https://ia.cr/2007/446>. 2007.

- [64] Carlos Aguilera Melchor and Yves Deswarte. "Single-Database Private Information Retrieval Schemes: Overview, Performance Study, and Usage with Statistical Databases". In: *Privacy in Statistical Databases*. Ed. by Josep Domingo-Ferrer and Luisa Franconi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 257–265. ISBN: 978-3-540-49332-7. DOI: 10.1007/11930242_22.
- [65] Rafail Ostrovsky and William E. Skeith. "A Survey of Single-Database Private Information Retrieval: Techniques and Applications". In: *Public Key Cryptography – PKC 2007*. Ed. by Tatsuaki Okamoto and Xiaoyun Wang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 393–411. ISBN: 978-3-540-71677-8. DOI: 10.1007/978-3-540-71677-8_26.
- [66] Christian Wieschebrink. "Two NP-complete Problems in Coding Theory with an Application in Code Based Cryptography". In: *IEEE International Symposium on Information Theory*. 2006, pp. 1733–1737. DOI: 10.1109/ISIT.2006.261651.
- [67] Radu Sion and Bogdan Carbuñar. "On the Computational Practicality of Private Information Retrieval". In: *Proceedings of the 14th Annual Network and Distributed System Security Symposium*. NDSS '07. San Diego, CA, USA: Internet Society, 2007, pp. 1–10.
- [68] Craig Gentry and Zulfikar Ramzan. "Single-Database Private Information Retrieval with Constant Communication Rate". In: *Automata, Languages and Programming*. Ed. by Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 803–815. ISBN: 978-3-540-31691-6. DOI: 10.1007/11523468_65.
- [69] Ryan Henry, Femi Olumofin, and Ian Goldberg. "Practical PIR for Electronic Commerce". In: *Proceedings of the 18th ACM Conference on Computer and Communications Security*. CCS '11. Chicago, Illinois, USA: Association for Computing Machinery, 2011, pp. 677–690. ISBN: 9781450309486. DOI: 10.1145/2046707.2046784.
- [70] Ian Goldberg. "Improving the Robustness of Private Information Retrieval". In: *IEEE Symposium on Security and Privacy (SP '07)*. 2007, pp. 131–148. DOI: 10.1109/SP.2007.23.
- [71] Femi Olumofin and Ian Goldberg. "Privacy-Preserving Queries over Relational Databases". In: *Privacy Enhancing Technologies*. Ed. by Mikhail J. Atallah and Nicholas J. Hopper. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 75–92. ISBN: 978-3-642-14527-8. DOI: 10.1007/978-3-642-14527-8_5.
- [72] Femi Olumofin, Piotr K. Tysowski, Ian Goldberg, and Urs Hengartner. "Achieving Efficient Query Privacy for Location Based Services". In: *Privacy Enhancing Technologies*. Ed. by Mikhail J. Atallah and Nicholas J. Hopper. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 93–110. ISBN: 978-3-642-14527-8. DOI: 10.1007/978-3-642-14527-8_6.
- [73] Ian Goldberg et al. *Percy++ / PIR in C++*. Sourceforge, accessed on 31 August 2022. <http://percy.sourceforge.net>. Oct. 2014.
- [74] Amos Beimel. "Private Information Retrieval: A Primer". In: *Department of Computer Science, Ben-Gurion University* (2008). <https://www.cs.bgu.ac.il/~beimel/Papers/PIRsurvey.pdf>.
- [75] William Gasarch. "A Survey on Private Information Retrieval". In: *Bulletin of the EATCS* 82.72-107 (2004), p. 113.
- [76] Adi Shamir. "How to Share a Secret". In: *Communications of the ACM* 22.11 (Nov. 1979), pp. 612–613. ISSN: 0001-0782. DOI: 10.1145/359168.359176.
- [77] Jiawei Han, Jian Pei, and Hanghang Tong. *Data Mining: Concepts and Techniques*. 4th ed. Morgan Kaufmann, July 2022. ISBN: 978-0-12-811760-6.
- [78] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. "From Data Mining to Knowledge Discovery in Databases". In: *AI Magazine* 17.3 (1996), pp. 37–37. DOI: 10.1609/aimag.v17i3.1230.

- [79] Nawal Mohammed Almutairi. "Privacy Preserving Third Party Data Mining Using Cryptography". PhD thesis. Liverpool, UK, Feb. 2020.
- [80] Justin Zhan. "Privacy-Preserving Collaborative Data Mining". In: *IEEE Computational Intelligence Magazine* 3.2 (2008), pp. 31–41. doi: 10.1109/MCI.2008.919071.
- [81] Xu Wei-Jiang, Huang Liu-Sheng, Luo Yong-Long, Yao Yi-Fei, and Jing Wei-Wei. "Privacy-Preserving DBSCAN Clustering Over Vertically Partitioned Data". In: *International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*. 2007, pp. 850–856. doi: 10.1109/MUE.2007.174.
- [82] Dongjie Jiang, Anrong Xue, Shiguang Ju, Weihe Chen, and Handa Ma. "Privacy-Preserving DBSCAN on Horizontally Partitioned Data". In: *IEEE International Symposium on IT in Medicine and Education*. 2008, pp. 1067–1072. doi: 10.1109/ITME.2008.4744034.
- [83] K. Anil Kumar and C. Pandu Rangan. "Privacy Preserving DBSCAN Algorithm for Clustering". In: *Advanced Data Mining and Applications*. Ed. by Reda Alhajj, Hong Gao, Jianzhong Li, Xue Li, and Osmar R. Zaïane. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 57–68. ISBN: 978-3-540-73871-8. doi: 10.1007/978-3-540-73871-8_7.
- [84] Mark Shaneck, Yongdae Kim, and Vipin Kumar. "Privacy Preserving Nearest Neighbor Search". In: *Machine Learning in Cyber Trust: Security, Privacy, and Reliability*. Boston, MA: Springer US, 2009, pp. 247–276. ISBN: 978-0-387-88735-7. doi: 10.1007/978-0-387-88735-7_10.
- [85] Yehuda Lindell and Benny Pinkas. "Privacy Preserving Data Mining." In: *Journal of Cryptology* 15.3 (2002), pp. 177–206. doi: 10.1007/s00145-001-0019-2.
- [86] Jinfei Liu, Joshua Zhexue Huang, Jun Luo, and Li Xiong. "Privacy Preserving Distributed DBSCAN Clustering". In: *Proceedings of the Joint EDBT/ICDT Workshops*. EDBT-ICDT '12. Berlin, Germany: Association for Computing Machinery, 2012, pp. 177–185. ISBN: 9781450311434. doi: 10.1145/2320765.2320819.
- [87] Qiuwei Tong, Xiu Li, and Bo Yuan. "Efficient Distributed Clustering using Boundary Information". In: *Neurocomputing* 275 (2018), pp. 2355–2366. ISSN: 0925-2312. doi: 10.1016/j.neucom.2017.11.014.
- [88] Deepti Mittal, Damandeep Kaur, and Ashish Aggarwal. "Secure Data Mining in Cloud Using Homomorphic Encryption". In: *IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*. 2014, pp. 1–7. doi: 10.1109/CCEM.2014.7015496.
- [89] Andrew C. Yao. "Protocols for Secure Computations". In: *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*. 1982, pp. 160–164. doi: 10.1109/SFCS.1982.38.
- [90] Zekeriya Erkin, Thijs Veugen, Tomas Toft, and Reginald L. Lagendijk. "Privacy-Preserving User Clustering in a Social Network". In: *First IEEE International Workshop on Information Forensics and Security (WIFS)*. 2009, pp. 96–100. doi: 10.1109/WIFS.2009.5386476.
- [91] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. "Zero-Knowledge from Secure Multiparty Computation". In: *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing*. STOC '07. San Diego, California, USA: Association for Computing Machinery, 2007, pp. 21–30. ISBN: 9781595936318. doi: 10.1145/1250790.1250794.
- [92] George Robert Blakley. "Safeguarding Cryptographic Keys". In: *Proceedings of the International Workshop on Managing Requirements Knowledge*. Los Alamitos, CA, USA: IEEE Computer Society, June 1979, pp. 313–318. doi: 10.1109/MARK.1979.8817296.
- [93] Dan Boneh and Matthew Franklin. "Efficient Generation of Shared RSA Keys". In: *Advances in Cryptology — CRYPTO '97*. Ed. by Burton S. Kaliski. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 425–439. ISBN: 978-3-540-69528-8. doi: 10.1007/BFb0052253.

- [94] Omkant Pandey and Yannis Rouselakis. "Property Preserving Symmetric Encryption". In: *Advances in Cryptology – EUROCRYPT 2012*. Ed. by David Pointcheval and Thomas Johansson. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 375–391. ISBN: 978-3-642-29011-4. doi: 10.1007/978-3-642-29011-4_23.
- [95] Carmit Hazay, Gert Læssøe Mikkelsen, Tal Rabin, Tomas Toft, and Angelo Agatino Nicolosi. "Efficient RSA Key Generation and Threshold Paillier in the Two-Party Setting". In: *Journal of Cryptology* 32.2 (2019), pp. 265–323. doi: 10.1007/s00145-017-9275-7.
- [96] Michael Beye, Zekeriya Erkin, and Reginald L. Lagendijk. "Efficient Privacy Preserving K-means Clustering in a Three-Party Setting". In: *IEEE International Workshop on Information Forensics and Security*. 2011, pp. 1–6. doi: 10.1109/WIFS.2011.6123148.
- [97] Fang-Yu Rao, Bharath K. Samanthula, Elisa Bertino, Xun Yi, and Dongxi Liu. "Privacy-Preserving and Outsourced Multi-user K-Means Clustering". In: *IEEE Conference on Collaboration and Internet Computing (CIC)*. 2015, pp. 80–89. doi: 10.1109/CIC.2015.20.
- [98] Mohammad Shahriar Rahman, Anirban Basu, and Shinsaku Kiyomoto. "Towards Outsourced Privacy-Preserving Multiparty DBSCAN". In: *IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC)*. 2017, pp. 225–226. doi: 10.1109/PRDC.2017.42.
- [99] Mohammed Golam Kaosar, Russell Paulet, and Xun Yi. "Fully Homomorphic Encryption Based Two-Party Association Rule Mining". In: *Data & Knowledge Engineering* 76-78 (2012), pp. 1–15. ISSN: 0169-023X. doi: 10.1016/j.ddata.2012.03.003.
- [100] Bharath K. Samanthula, Yousef Elmehdwi, and Wei Jiang. "k-Nearest Neighbor Classification over Semantically Secure Encrypted Relational Data". In: *IEEE Transactions on Knowledge and Data Engineering* 27.5 (2015), pp. 1261–1273. doi: 10.1109/TKDE.2014.2364027.
- [101] Nathan Dowlin et al. "CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy". In: *Proceedings of the 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. New York, New York, USA: PMLR, June 2016, pp. 201–210.
- [102] George Drosatos and Pavlos S. Efraimidis. "An Efficient Privacy-Preserving Solution for Finding the Nearest Doctor". In: *Personal and Ubiquitous Computing* 18.1 (2014), pp. 75–90. ISSN: 1617-4909. doi: 10.1007/s00779-012-0619-x.
- [103] George Drosatos, Pavlos S. Efraimidis, Ioannis N. Athanasiadis, Matthias Stevens, and Ellie D'Hondt. "Privacy-Preserving Computation of Participatory Noise Maps in the Cloud". In: *Journal of Systems and Software* 92 (2014), pp. 170–183. ISSN: 0164-1212. doi: 10.1016/j.jss.2014.01.035.
- [104] Terrence J. Sejnowski and Charles R. Rosenberg. "Parallel Networks that Learn to Pronounce English Text". In: *Complex Systems* 1.1 (1987), pp. 145–168.
- [105] Paul Werbos. "Advanced Forecasting Methods for Global Crisis Warning and Models of Intelligence". In: *General Systems: Yearbook of the Society for General Systems Research* XXII (1977), pp. 25–38.
- [106] Kai Labusch, Erhardt Barth, and Thomas Martinetz. "Simple Method for High-Performance Digit Recognition Based on Sparse Coding". In: *IEEE Transactions on Neural Networks* 19.11 (2008), pp. 1985–1989. doi: 10.1109/TNN.2008.2005830.
- [107] Ehsan Hesamifard, Hassan Takabi, Mehdi Ghasemi, and Catherine Jones. "Privacy-Preserving Machine Learning in Cloud". In: *Proceedings of the 2017 on Cloud Computing Security Workshop*. CCSW '17. Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 39–43. ISBN: 9781450352048. doi: 10.1145/3140649.3140655.

- [108] Ehsan Hesamifard, Hassan Takabi, Mehdi Ghasemi, and Rebecca N. Wright. "Privacy-Preserving Machine Learning as a Service". In: *Proceedings on Privacy Enhancing Technologies*. Vol. 2018. 3. 2018, pp. 123–142. doi: 10.1515/popets-2018-0024.
- [109] Harry Chandra Tanuwidjaja, Rakyong Choi, Seunggeun Baek, and Kwangjo Kim. "Privacy-Preserving Deep Learning on Machine Learning as a Service—a Comprehensive Survey". In: *IEEE Access* 8 (2020), pp. 167425–167447. issn: 2169-3536. doi: 10.1109/ACCESS.2020.3023084.
- [110] Thore Graepel, Kristin Lauter, and Michael Naehrig. "ML Confidential: Machine Learning on Encrypted Data". In: *Information Security and Cryptology – ICISC 2012*. Ed. by Taekyoung Kwon, Mun-Kyu Lee, and Daesung Kwon. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 1–21. isbn: 978-3-642-37682-5. doi: 10.1007/978-3-642-37682-5_1.
- [111] Marina Skurichina and Robert P.W. Duin. "Bagging, Boosting and the Random Subspace Method for Linear Classifiers". In: *Pattern Analysis & Applications* 5.2 (2002), pp. 121–135. doi: 10.1007/s100440200011.
- [112] Seung-Jean Kim, Alessandro Magnani, and Stephen Boyd. "Robust Fisher Discriminant Analysis". In: *Advances in Neural Information Processing Systems*. Ed. by Y. Weiss, B. Schölkopf, and J. Platt. Vol. 18. MIT Press, 2005.
- [113] Joppe W. Bos, Kristin Lauter, Jake Loftus, and Michael Naehrig. "Improved Security for a Ring-Based Fully Homomorphic Encryption Scheme". In: *Cryptography and Coding*. Ed. by Martijn Stam. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 45–64. isbn: 978-3-642-45239-0. doi: 10.1007/978-3-642-45239-0_4.
- [114] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, Lihua Wang, and Shihō Moriai. "Privacy-Preserving Deep Learning via Additively Homomorphic Encryption". In: *IEEE Transactions on Information Forensics and Security* 13.5 (2018), pp. 1333–1345. doi: 10.1109/TIFS.2017.2787987.
- [115] Reza Shokri and Vitaly Shmatikov. "Privacy-Preserving Deep Learning". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. CCS ’15. Denver, Colorado, USA: Association for Computing Machinery, 2015, pp. 1310–1321. isbn: 9781450338325. doi: 10.1145/2810103.2813687.
- [116] Amartya Sanyal, Matt Kusner, Adria Gascon, and Varun Kanade. "TAPAS: Tricks to Accelerate (encrypted) Prediction As a Service". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. PMLR, July 2018, pp. 4490–4499.
- [117] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. "Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds". In: *Advances in Cryptology – ASIACRYPT 2016*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 3–33. isbn: 978-3-662-53887-6. doi: 10.1007/978-3-662-53887-6_1.
- [118] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. "Quantized Neural Networks: Training Neural Networks with Low Precision Weights and Activations". In: *Journal of Machine Learning Research* 18.187 (2018). <http://jmlr.org/papers/v18/16-456.html>, pp. 1–30.
- [119] Florian Bourse, Michele Minelli, Matthias Minihold, and Pascal Paillier. "Fast Homomorphic Evaluation of Deep Discretized Neural Networks". In: *Advances in Cryptology – CRYPTO 2018*. Ed. by Hovav Shacham and Alexandra Boldyreva. Cham: Springer International Publishing, 2018, pp. 483–512. isbn: 978-3-319-96878-0. doi: 10.1007/978-3-319-96878-0_17.

- [120] Xiaoqian Jiang, Miran Kim, Kristin Lauter, and Yongsoo Song. "Secure Outsourced Matrix Computation and Application to Neural Networks". In: *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*. CCS '18. Toronto, Canada: Association for Computing Machinery, 2018, pp. 1209–1222. ISBN: 978-1-45-035693-0. DOI: 10.1145/3243734.3243837.
- [121] Payman Mohassel and Yupeng Zhang. "SecureML: A System for Scalable Privacy-Preserving Machine Learning". In: *IEEE Symposium on Security and Privacy (SP)*. 2017, pp. 19–38. DOI: 10.1109/SP.2017.12.
- [122] Jian Liu, Mika Juuti, Yao Lu, and N. Asokan. "Oblivious Neural Network Predictions via Min-iONN Transformations". In: *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*. CCS '17. Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 619–631. ISBN: 9781450349468. DOI: 10.1145/3133956.3134056.
- [123] Payman Mohassel and Peter Rindal. "ABY³: A Mixed Protocol Framework for Machine Learning". In: *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*. CCS '18. Toronto, Canada: Association for Computing Machinery, 2018, pp. 35–52. ISBN: 978-1-45-035693-0. DOI: 10.1145/3243734.3243760.
- [124] M. Sadegh Riazi et al. "Chameleon: A Hybrid Secure Computation Framework for Machine Learning Applications". In: *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*. ASIACCS '18. Incheon, Republic of Korea: Association for Computing Machinery, 2018, pp. 707–721. ISBN: 9781450355766. DOI: 10.1145/3196494.3196522.
- [125] Sameer Wagh, Divya Gupta, and Nishanth Chandran. "SecureNN: 3-Party Secure Computation for Neural Network Training". In: *Proceedings on Privacy Enhancing Technologies*. Vol. 2019. 3. 2019, pp. 26–49. DOI: 10.2478/popets-2019-0035.

Μέρος IV

ΕΙΔΙΚΑ ΘΕΜΑΤΑ

ΚΕΦΑΛΑΙΟ 13

ΕΛΑΦΡΑ ΚΡΥΠΤΟΓΡΑΦΙΑ

Περίληψη

Η Ελαφρά Κρυπτογραφία (Lightweight Cryptography) είναι ένα πεδίο της Κρυπτογραφίας που αναπτύχθηκε τα τελευταία χρόνια για να σχεδιάσει τεχνικές για συσκευές που έχουν περιορισμό σε πόρους, όπως ενέργειας, συνδεσιμότητας, υλικού και λογισμικού, και τις οποίες συναντάμε κυρίως στο Διαδίκτυο των Πραγμάτων (Internet of Things – IoT). Η χρήση αλγορίθμων γενικού σκοπού σε τέτοια συστήματα, όπως αυτών που παρουσιάστηκαν στα Κεφάλαια 1 και 3, καθίσταται αναποτελεσματική. Για αυτό το σκοπό έχει προταθεί και αναπτυχθεί μια πληθώρα αλγορίθμων ελαφράς κρυπτογραφίας οι οποίοι εξυπηρετούν αυτές τις ανάγκες. Πιο αναλυτικά, στην Ενότητα 13.1 γίνεται μια εισαγωγή στην ελαφρά κρυπτογραφία, αναφέροντας τους λόγους εμφάνισής της, αλλά και τις σημαντικότερες προσπάθειες που έχουν γίνει προς αυτήν την κατεύθυνση. Στην Ενότητα 13.2 παρουσιάζονται τα βασικά χαρακτηριστικά των αλγορίθμων ελαφράς κρυπτογραφίας, όσον αφορά την χρήση τους, αλλά και τις προδιαγραφές που πρέπει να καλύπτουν. Στην Ενότητα 13.3 παρουσιάζονται αναλυτικά οι πέντε πιο αποδοτικοί αλγόριθμοι από αυτούς που προκρίθηκαν στον 3ο γύρο δοκιμών του NIST [1], δηλαδή τους ASCON, GIFT-COFB, SPARKLE, TinyJAMBU και Xoodyak, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών αρχών της συμμετρικής κρυπτογράφησης (Κεφάλαιο 1) και των συναρτήσεων σύνοψης (Κεφάλαιο 3).

13.1 Εισαγωγή

Σε περιβάλλοντα με περιορισμένους υπολογιστικούς πόρους, όπως στο Διαδίκτυο των Πραγμάτων (IoT), η μαζική συνδεσιμότητα των συσκευών και ο τεράστιος όγκος δεδομένων που ανταλλάσσονται, σε πολλές περιπτώσεις περιλαμβάνοντας προσωπικά και ιδιαίτερης αξίας δεδομένα, καθιστούν ιδιαίτερα ελκυστικές αυτού του είδους συσκευές σε διάφορους τύπους επιθέσεων. Για παράδειγμα, τέτοιες συσκευές θα μπορούσαν να είναι αισθητήρες που συλλέγουν δεδομένα από το περιβάλλον μιας βιομηχανίας ή ενός ασθενούς και οι οποίοι μεταφέρουν αυτά τα δεδομένα σε απομακρυσμένες τοποθεσίες για περαιτέρω επεξεργασία. Η προστα-

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

σία όλων αυτών των δεδομένων σε πολλές περιπτώσεις αποτελεί μονόδρομο. Οι συσκευές IoT έχουν σημαντικούς περιορισμούς αναφορικά με τους διαθέσιμους πόρους και κατ' επέκταση τη χρήση αυτών, όπως είναι η χωρητικότητα αποθήκευσης και η κατανάλωση ενέργειας. Λόγω της περιορισμένης υπολογιστικής ισχύος, της περιορισμένης μνήμης και των πόρων ενεργειακής αυτονομίας των συσκευών, είναι δύσκολο να χρησιμοποιηθούν παραδοσιακοί κρυπτογραφικοί αλγόριθμοι για την ασφάλεια των πληροφοριών. Για τον λόγο αυτό, απαιτούνται πιο αποτελεσματικοί αλγόριθμοι που μπορούν να ανταποκριθούν σε αυτούς τους περιορισμένους πόρους. Τη λύση σε αυτό το πρόβλημα έρχονται να δώσουν οι αλγόριθμοι ελαφράς κρυπτογραφίας.

Η ελαφρά κρυπτογραφία είναι μια υποπεριοχή της κρυπτογραφίας που στοχεύει στην παροχή λύσεων προσαρμοσμένων σε συσκευές περιορισμένων πόρων. Σε αυτή τη βάση έχει γίνει σημαντική δουλειά από την ακαδημαϊκή κοινότητα και τη βιομηχανία, περιλαμβάνοντας αποτελεσματικές εφαρμογές συμβατικών αλγορίθμων κρυπτογραφίας και το σχεδιασμό και ανάλυση νέων ελαφρών αλγορίθμων. Στο πλαίσιο αυτό, το NIST (National Institute of Standards and Technology) των ΗΠΑ ξεκίνησε το 2015 ένα πρόγραμμα για να μελετήσει την απόδοση σε συσκευές περιορισμένων πόρων ενός συνόλου αλγορίθμων. Στις ενότητες που ακολουθούν αναλύονται οι αλγόριθμοι που προκρίθηκαν στον 3ο γύρο δοκιμών του NIST [1], δηλαδή οι: ASCON, GIFT-COFB, SPARKLE, TinyJAMBU, και Xoodyak.

13.2 Κατηγορίες και Χαρακτηριστικά

Ένα από τα πιο βασικά χαρακτηριστικά των αλγορίθμων ελαφράς κρυπτογραφίας, είναι ο τύπος τους, που προσδιορίζει και τη βασική τους χρήση. Έτσι λοιπόν, οι τύποι στους οποίους διακρίνονται είναι:

- Αλγόριθμοι κρυπτογράφησης μπλοκ
- Αλγόριθμοι κρυπτογράφησης ροής
- Αλγόριθμοι συναρτήσεων σύνοψης
- Αλγόριθμοι πιστοποιημένης κρυπτογράφησης με συσχετισμένα δεδομένα (Authenticated Encryption with Associated Data – AEAD)

Επιπλέον των παραπάνω τύπων, τα ακόλουθα χαρακτηριστικά είναι αυτά που καθορίζουν τις προδιαγραφές των αλγορίθμων ελαφράς κρυπτογραφίας:

- **Πολυπλοκότητα υλοποίησης:** Πρόκειται για έναν από τους σημαντικότερους παράγοντες που επηρεάζουν τον σχεδιασμό ενός αλγορίθμου και καθορίζεται από το πλήθος των λογικών πυλών που απαιτούνται για την υλοποίησή του. Εκτός από την πολυπλοκότητα υλοποίησης, το πλήθος των λογικών πυλών αντικατοπτρίζει ένα τμήμα της περιοχής του ολοκληρωμένου κυκλώματος που καταλαμβάνεται από την υλοποίηση του αλγορίθμου στο υλικό.
- **Ταχύτητα επεξεργασίας δεδομένων:** Η ταχύτητα επεξεργασίας των δεδομένων χαρακτηρίζεται από τα ακόλουθα δύο στοιχεία:
 - **Καθυστέρηση (Latency):** Καθορίζει το χρόνο που απαιτείται για την εκτέλεση μιας διαδικασίας (π.χ. κρυπτογράφησης ή αποκρυπτογράφησης).
 - **Ταχύτητα (Throughput):** Αναφέρεται στην ποσότητα δεδομένων που επεξεργάζεται το κύκλωμα σε μια μονάδα χρόνου και μετριέται σε bits ή bytes ανά δευτερόλεπτο.
- **Μέγεθος μνήμης RAM:** Αφορά στην ποσότητα των δεδομένων που γράφονται στη μνήμη RAM κατά την εκτέλεση του αλγορίθμου.

- Ταχύτητα υλοποίησης:** Η αύξηση της ταχύτητας υλοποίησης μπορεί να επιτευχθεί με τη μείωση του αριθμού των κύκλων εκτέλεσης. Μερικές φορές ακόμη και η απόδοση χρησιμοποιείται ως μια παράμετρος που εκφράζει τη ταχύτητα υλοποίησης και πρέπει να είναι και αυτή αρκετά υψηλή.
- Κατανάλωση ενέργειας:** Για συσκευές με περιορισμένους ενεργειακούς πόρους, όπως αυτές που τροφοδοτούνται από μπαταρία, είναι απαραίτητο η κατανάλωση ενέργειας να διατηρείται σε χαμηλά επίπεδα. Η μέτρησή της πραγματοποιείται με βάση τον αριθμό των κύκλων εκτέλεσης ή η ένταση του ρεύματος που διαρρέει το κύκλωμα.

Άλλα χαρακτηριστικά που προσδιορίζουν τους αλγορίθμους και αφορούν την υλοποίησή τους σε υλικό, είναι τα ακόλουθα:

- Μέγεθος κλειδιού:** Αφορά κατά κύριο λόγο τους αλγόριθμους κρυπτογράφησης μπλοκ και ροής και καθορίζει το μέγεθος κλειδιών που χρησιμοποιούν (συνήθως είναι 128 bits).
- Μέγεθος μπλοκ:** Προσδιορίζει το μέγεθος του μπλοκ που υποστηρίζει ο αλγόριθμος κρυπτογράφησης (συνήθως είναι 128 bits).
- Ισοδύναμο πύλης (Gate Equivalent – GE):** Αποτελεί μια μονάδα μέτρησης του πλήθους των λογικών πυλών που απαιτούνται για την υλοποίηση του αλγορίθμου. Είναι ανεξάρτητη από την τεχνολογία υλικού που χρησιμοποιείται και αντικατοπτρίζει την πολυπλοκότητα κατασκευής των ψηφιακών ηλεκτρονικών κυκλωμάτων (συνήθως ≤ 3000 GE). Στον Πίνακα 13.1 [2] παρουσιάζεται ενδεικτικά μια κατηγοριοποίηση των υλοποιήσεων υλικού με βάση αυτή την μονάδα μέτρησης.

Πίνακας 13.1: Κατηγορίες υλοποίησης υλικού με την μονάδα μέτρησης Gate Equivalent (GE).

Κατηγορία	Gate Equivalent (GE)
Υπερ-ελαφρά υλοποίηση	≤ 1000
Χαμηλού κόστους υλοποίηση	> 1000 και ≤ 2000
Ελαφρά υλοποίηση	> 2000 και ≤ 3000

13.3 Αλγόριθμοι Ελαφράς Κρυπτογραφίας

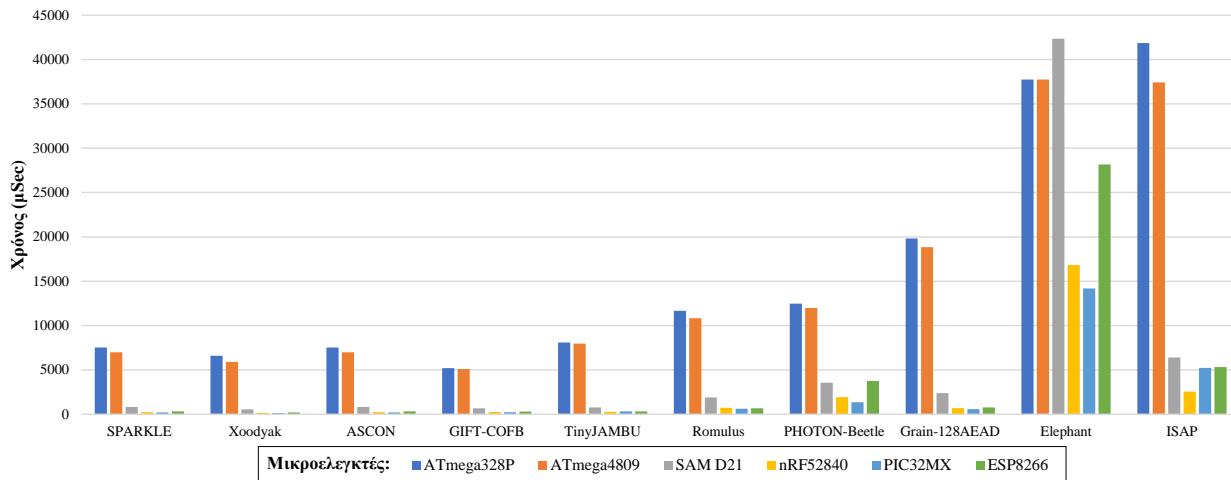
Στον Πίνακα 13.2 γίνεται μια συγκριτική παρουσίαση των αλγορίθμων ελαφράς κρυπτογραφίας που προκρίθηκαν στον 3ο γύρο δοκιμών του NIST [1], σε μια προσπάθεια προτυποποίησης για την επιλογή ενός ή περισσότερων σχημάτων με λειτουργίες πιστοποιημένης κρυπτογράφησης με συσχετισμένα δεδομένα (AEAD) και με λειτουργίες σύνοψης, κατάλληλων για περιβάλλοντα με περιορισμένους υπολογιστικούς πόρους. Όπως γίνεται αντίληπτό, οι περισσότεροι (6/10) από αυτούς τους αλγορίθμους έχουν ως αρχή λειτουργίας τους την αντιμετάθεση (permutation), τρεις βασίζονται σε κρυπτογράφηση μπλοκ, και μόνο ένας σε κρυπτογράφηση ροής. Αντίστοιχα, για καθέναν από αυτούς τους αλγορίθμους, αναφέρεται η λειτουργία που μπορεί να παρέχουν, δηλαδή εάν μπορούν να χρησιμοποιηθούν για λειτουργίες σύνοψης (5/10), ή/και πιστοποιημένης κρυπτογράφησης με συσχετισμένα δεδομένα (AEAD, 10/10).

Επιπρόσθετα, όσον αφορά την κατάταξη των αλγορίθμων, αυτή πραγματοποιήθηκε με βάση την απόδοση που είχαν σε επεξεργαστές 8-bit AVR (ATmega2560 16 MHz) και 32-bit (ARM Cortex M3 και ESP32 Arduino) [3]. Με βάση αυτή την σειρά κατάταξης, είναι ξεκάθαρο ότι η οικογένεια αλγορίθμων SPARKLE επιτυγχάνει την καλύτερη απόδοση τόσο σε λειτουργίες σύνοψης όσο και AEAD. Ακολουθούν οι οικογένειες Xoodyak και ASCON, με την πρώτη να φαίνεται να έχει καλύτερη απόδοση σε 32-bit επεξεργαστές,

ενώ η δεύτερη να μην φαίνεται να παρουσιάζει σημαντικές διαφορές στις διαφορετικές αρχιτεκτονικές επεξεργαστών. Στην τέταρτη και πέμπτη θέση βρίσκονται οι αλγόριθμοι GIFT-COFB και TinyJAMBU λαμβάνοντας κυρίως υπόψη την απόδοσή τους στη λειτουργία AEAD, με τους υπόλοιπους να ακολουθούν με την εξής σειρά κατάταξης: Romulus, PHOTON-Beetle, Grain-128AEAD, Elephant, και ISAP. Στις υποενότητες που ακολουθούν γίνεται παρουσίαση των πέντε από τους 10 αυτούς αλγορίθμους που είχαν την καλύτερη κατάταξη απόδοσης συνολικά. Επίσης, στην επιλογή των ίδιων αλγορίθμων οδηγούμαστε λαμβάνοντας υπόψη το Σχήμα 13.1 που παρουσιάζει συγκριτικά τους αλγορίθμους όσον αφορά το χρόνο εκτέλεσης των πιο γρήγορων υλοποίησεων των κύριων παραλλαγών AEAD σε διάφορους μικροελεγκτές (οι δύο πρώτοι είναι αρχιτεκτονικής 8-bit AVR και οι υπόλοιποι τέσσερις είναι 32-bit).

Πίνακας 13.2: Συγκριτική παρουσίαση των φιναλίστ αλγορίθμων ελαφράς κρυπτογραφίας του NIST (η κατάταξη απόδοσης όσον αφορά τις λειτουργίες σύνοψης και AEAD έγινε με βάση το [3] για AVR και 32-bit επεξεργαστές).

Αλγόριθμος	Αρχή Λειτουργίας	Παρεχόμενες Λειτουργίες Σύνοψη	Κατάταξη Απόδοσης AVR CPUs	Κατάταξη Απόδοσης 32-bit CPUs	Κατάταξη Απόδοσης AEAD AVR CPUs	Κατάταξη Απόδοσης AEAD 32-bit CPUs
ASCON [4]	Αντιμετάθεση	+	+	4	3	3
Elephant [5]	Αντιμετάθεση		+		9	10
GIFT-COFB [6]	Κρυπτογράφηση μπλοκ		+		2	5
Grain-128AEAD [7]	Κρυπτογράφηση ροής		+		8	6
ISAP [8]	Αντιμετάθεση		+		10	9
PHOTON-Beetle [9]	Αντιμετάθεση	+	+	5	4	7
Romulus [10]	Κρυπτ. μπλοκ (tweakable)	+	+	2	4	6
SPARKLE [11]	Αντιμετάθεση	+	+	1	1	1
TinyJAMBU [12]	Κρυπτογράφηση μπλοκ		+		4	4
Xoodyak [13]	Αντιμετάθεση	+	+	2	1	1



Σχήμα 13.1: Χρόνος εκτέλεσης των πιο γρήγορων υλοποίησεων των κύριων παραλλαγών AEAD για πιστοποιημένη κρυπτογράφηση μηνύματος 16-byte και συσχετισμένων δεδομένων 16-byte σε διάφορους μικροελεγκτές [1].

13.3.1 Αλγόριθμος ASCON

Η κρυπτογραφική σούίτα ASCON [4] βασίζεται στην αντιμετάθεση και παρέχει πιστοποιημένη κρυπτογράφηση με συσχετισμένα δεδομένα (AEAD) και λειτουργία συνάρτησης σύνοψης. Η σούίτα αποτελείται από τους πιστοποιημένους αλγορίθμους κρυπτογράφησης ASCON-128 και ASCON-128a που αναδείχθηκαν στα πλαίσια του διαγωνισμού CAESAR (2014-2019). Επιπλέον, περιλαμβάνονται συναρτήσεις σύνοψης με

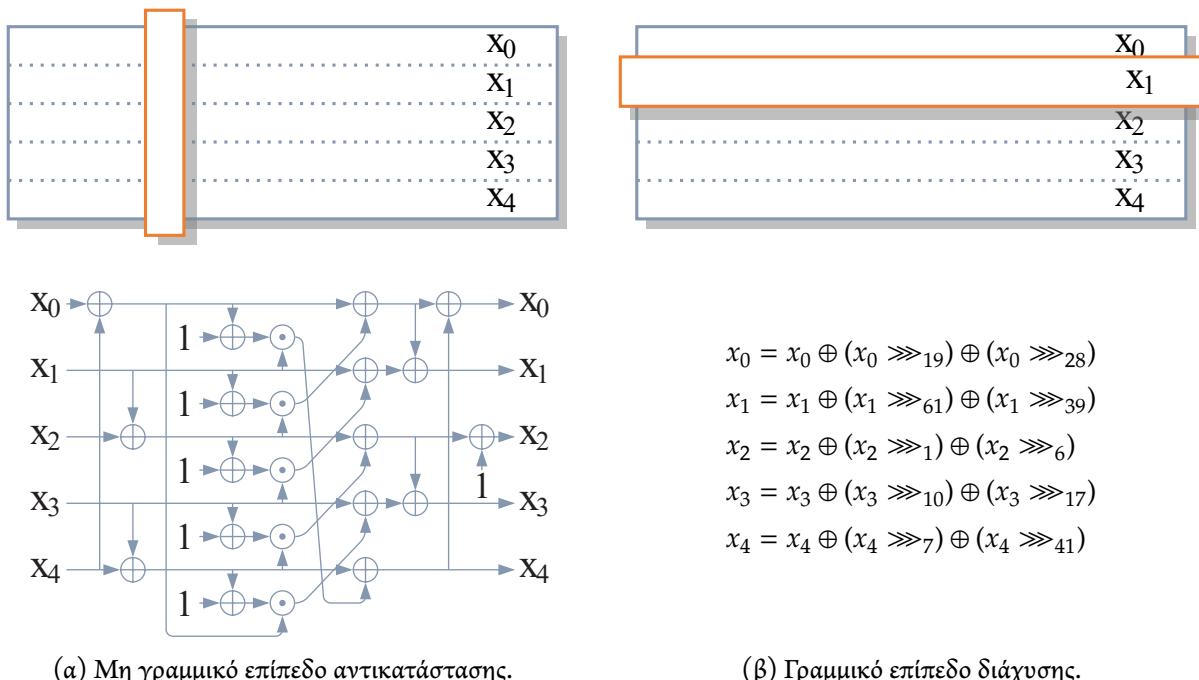
ονομασίες ASCON-HASH και ASCON-HASHA, καθώς και συναντήσεις σύνοψης μεταβλητής εξόδου με ονομασίες ASCON-XOF και ASCON-XOFA. Όλα τα σχήματα παρέχουν ασφάλεια της τάξης των 128-bit και χρησιμοποιούν εσωτερικά την ίδια αντιμετάθεση 320-bit (με διαφορετικό αριθμό γύρων), έτσι ώστε μια μικρή αλλαγή παραμέτρων να είναι αρκετή για την υλοποίηση τόσο μιας πιστοποιημένης κρυπτογράφησης AEAD όσο και μιας συνάρτησης σύνοψης.

Στις υποενότητες που ακολουθούν παρουσιάζονται τα εξής τρία βασικά στοιχεία της κρυπτογραφικής συνίτιας ASCON: η αντιμετάθεση (permutation) ASCON, οι λειτουργίες πιστοποιημένης κρυπτογράφησης ASCON, και οι λειτουργίες συναρτήσεων σύνοψης ASCON.

13.3.1.1 Αντιμετάθεση ASCON

Όλα τα μέλη της κρυπτογραφικής συνίτιας ASCON χρησιμοποιούν την ίδια ελαφρά αντιμετάθεση (permutation). Αυτή η αντιμετάθεση εφαρμόζει επαναληπτικά έναν γύρο μετασχηματισμών που βασίζεται σε ένα δίκτυο αντικαταστάσεων-αντιμετάθεσεων (Substitution-Permutation Network – SPN). Ο αριθμός των γύρων είναι $a = 12$ για την συνάρτηση αντιμετάθεσεων p^a και $b \in \{6, 8\}$ για την συνάρτηση αντιμετάθεσεων p^b . Ο κάθε γύρος μετασχηματισμών αποτελείται από τα ακόλουθα τρία βήματα που εφαρμόζονται στην εσωτερική κατάσταση των 320-bit χωρισμένη σε 5 λέξεις x_0, x_1, x_2, x_3, x_4 των 64-bit η καθεμία (Σχήμα 13.2):

- **Πρόσθεση σταθερών των γύρων:** Πραγματοποίηση της πράξης XOR (\oplus) μεταξύ μιας σταθεράς 1 byte (διαφορετική για κάθε γύρο) και της λέξης x_2 .
- **Μη γραμμικό επίπεδο αντικατάστασης:** Εφαρμογή ενός S-box αντικατάστασης των 5-bit κάθετα σε όλες τις λέξεις και για 64 φορές παράλληλα (Σχήμα 13.2α, όπου \odot υποδηλώνει την πράξη AND).
- **Γραμμικό επίπεδο διάχυσης:** Πραγματοποίηση της πράξης XOR (\oplus) σε διαφορετικά ολισθημένα (όπου \ggg_s υποδηλώνει μια ολίσθηση προς τα δεξιά κατά s θέσεις bit) αντίγραφα δεδομένων της κάθε λέξης οριζόντια (Σχήμα 13.2β).



Σχήμα 13.2: Γραμμικός και μη γραμμικός μετασχηματισμός της αντιμετάθεσης ASCON.

13.3.1.2 Λειτουργίες Πιστοποιημένης Κρυπτογράφησης ASCON

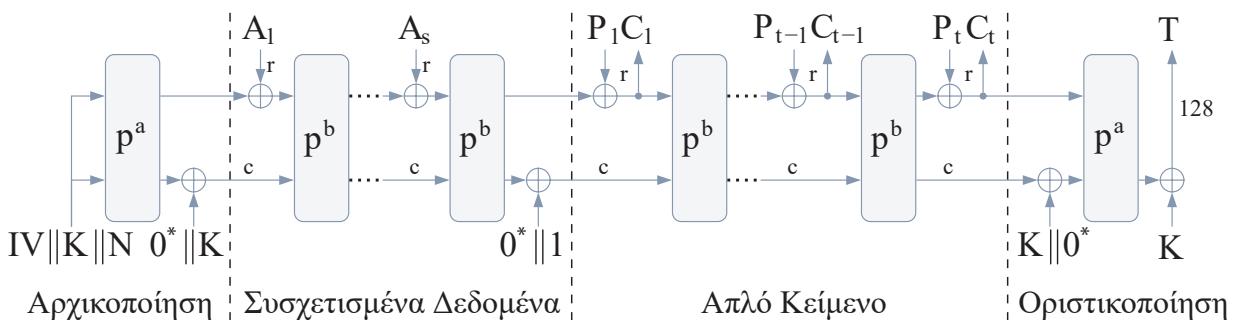
Οι λειτουργίες πιστοποιημένης κρυπτογράφησης ASCON έχουν ως βασικό δομικό τους στοιχείο μια διπλή κατασκευή Sponge (βλέπε Ενότητα 3.2.4). Τα μέλη της οικογένειας πιστοποιημένης κρυπτογράφησης ASCON έχουν ως παραμέτρους το μήκος του κλειδιού $k \leq 160$ -bit, τον ρυθμό δεδομένων r (που αντιστοιχεί στο μέγεθος του μπλοκ δεδομένων) και το πλήθος a και b των εσωτερικών γύρων. Η πιστοποιημένη κρυπτογράφηση αποτελείται από τον αλγόριθμο κρυπτογράφησης $E_{k,r,a,b}$ και τον αλγόριθμο αποκρυπτογράφησης $D_{k,r,a,b}$. Η διαδικασία πιστοποιημένης κρυπτογράφησης $E_{k,r,a,b}$ δέχεται ως είσοδο ένα μυστικό κλειδί K μεγέθους k -bit, έναν μοναδικό αριθμό (nonce) N μεγέθους 128-bit, τα συσχετισμένα δεδομένα A αυθαίρετου μεγέθους και ένα απλό κείμενο (plaintext) P αυθαίρετου μεγέθους. Το αποτέλεσμα εξόδου αποτελείται από το πιστοποιημένο κρυπτοκείμενο (ciphertext) C ακριβώς ίδιου μεγέθους με το απλό κείμενο P και επιπλέον από μια ετικέτα πιστοποίησης T μεγέθους 128-bit, η οποία πιστοποιεί την αυθεντικότητα τόσο των συσχετισμένων δεδομένων όσο και του κρυπτογραφημένου μηνύματος:

$$E_{k,r,a,b}(K, N, A, P) = (C, T)$$

Πιο αναλυτικά, η διαδικασία κρυπτογράφησης χωρίζεται σε τέσσερις φάσεις (Σχήμα 13.3):

- **Αρχικοποίηση:** Αρχικοποιεί την εσωτερική κατάσταση με το μυστικό κλειδί K και έναν μοναδικό αριθμό (nonce) N .
- **Επεξεργασία συσχετισμένων δεδομένων:** Ενημερώνει την κατάσταση με το μπλοκ συσχετισμένων δεδομένων A_i .
- **Επεξεργασία απλού κειμένου:** Εισάγει τα μπλοκ δεδομένων του απλού κειμένου P_i στην κατάσταση και εξάγει τα μπλοκ δεδομένων του κρυπτογραφημένου κειμένου C_i .
- **Οριστικοποίηση:** Εισάγει ξανά το μυστικό κλειδί K και εξάγει την ετικέτα T για τον έλεγχο της αυθεντικότητας.

Μετά από κάθε μπλοκ δεδομένων που εισάγεται (εκτός από το τελευταίο μπλοκ του απλού κειμένου), εφαρμόζεται η συνάρτηση αντιμεταθέσεων p^b στην πλήρη εσωτερική κατάσταση, ενώ κατά την αρχικοποίηση και την οριστικοποίηση, χρησιμοποιείται η συνάρτηση αντιμεταθέσεων p^a που διαθέτει μεγαλύτερο πλήθος γύρων.

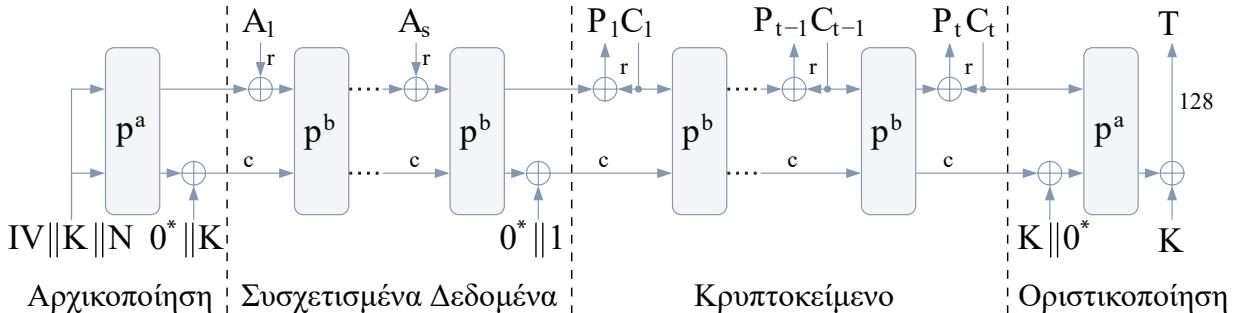


Σχήμα 13.3: Διαδικασία κρυπτογράφησης στην κρυπτογραφική σουίτα ASCON.

Αντίστοιχα, η διαδικασία αποκρυπτογράφησης και επαλήθευσης $D_{k,r,a,b}$ δέχεται ως είσοδο το μυστικό κλειδί K , τον ίδιο μοναδικό αριθμό (nonce) N , τα συσχετισμένα δεδομένα A , το κρυπτογραφημένο κείμενο C και την ετικέτα T . Το αποτέλεσμα εξόδου αποτελείται είτε από το απλό κείμενο P εάν η επαλήθευση της ετικέτας είναι σωστή, είτε από ένα σφάλμα \perp εάν η επαλήθευση της ετικέτας αποτύχει:

$$D_{k,r,a,b}(K, N, A, C, T) \in \{P, \perp\}$$

Η αναλυτική διαδικασία αποκρυπτογράφησης παρουσιάζεται στο Σχήμα 13.4 και ακολουθεί ανάλογες φάσεις με την διαδικασία κρυπτογράφησης. Η κύρια διαφορά της επικεντρώνεται στην επεξεργασία κρυπτοκειμένου, όπου αυτή την φορά έχει ως είσοδο στην εσωτερική κατάσταση τα δεδομένων του κρυπτοκειμένου C_i και ως έξοδο τα μπλοκ δεδομένων του απλού κειμένου P_i .



Σχήμα 13.4: Διαδικασία αποκρυπτογράφησης στην κρυπτογραφική σουίτα ASCON.

Οι αριθμοί των γύρων a και b , καθώς και ο ρυθμός r και η χωρητικότητα c της κατασκευής Sponge, εξαρτώνται από την παραλλαγή κρυπτογράφησης του ASCON (ASCON-128 και ASCON-128a). Οι προτεινόμενες παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης ASCON συνοψίζονται στον Πίνακα 13.3.

Πίνακας 13.3: Παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης ASCON.

Αλγόριθμος	Μέγεθος (bits)					Γύροι	
	Κλειδί	Nonce	Επικέτα	Ρυθμός	Χωρητικότητα	p^a	p^b
ASCON-128	128	128	128	64	256	12	6
ASCON-128a	128	128	128	128	192	12	8

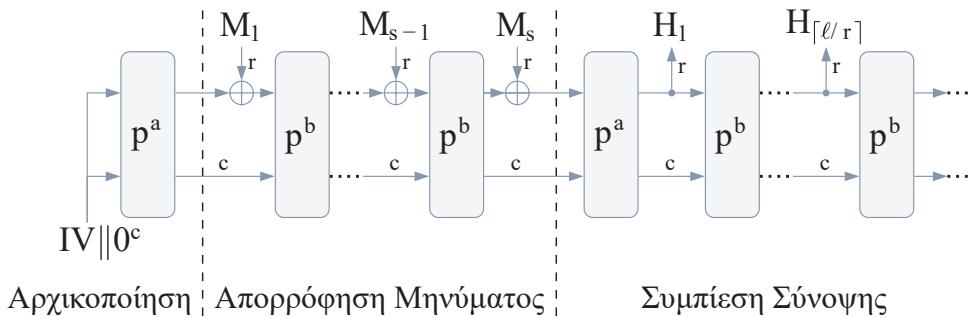
13.3.1.3 Λειτουργίες Συναρτήσεων Σύνοψης ASCON

Οι λειτουργίες συναρτήσεων σύνοψης ASCON έχουν και αυτές ως βασικό δομικό τους στοιχείο μια κατασκευή Sponge (βλ. Ενότητα 3.2.4), μόνο που σε αυτή την περίπτωση δεν είναι διπλή. Η οικογένεια συναρτήσεων σύνοψης (ASCON-HASH και ASCON-HASHA) καθώς και η οικογένεια συναρτήσεων επεκτάσιμης εξόδου (ASCON-XOF και ASCON-XOFA) ορίζονται χρησιμοποιώντας την συνάρτηση εξόδου $X_{h,r,a,b}$ και έχοντας ως παραμέτρους τον ρυθμό δεδομένων r (αντιστοιχεί στο μέγεθος του μπλοκ δεδομένων), το πλήθος a και b των εσωτερικών γύρων, και το όριο του μήκους εξόδου h (για $h = 0$, έχει απεριόριστο μήκος εξόδου). Η συνάρτηση $X_{h,r,a,b}$ αντιστοιχίζει ένα μήνυμα M αυθαίρετου μήκους σε μια έξοδο σύνοψης H αυθαίρετα καθορισμένου μήκους ℓ ($\leq h$):

$$X_{h,r,a,b}(M, \ell) = H$$

Η αναλυτική διαδικασία δημιουργίας σύνοψης παρουσιάζεται στο Σχήμα 13.5. Οι λειτουργίες σύνοψης ASCON απορροφούν (absorb) ένα μήνυμα M σε μπλοκ δεδομένων M_i μεγέθους 64-bit, ενώ κατόπιν συμπιέζουν (squeeze) την τιμή σύνοψης H σε μπλοκ δεδομένων H_i μεγέθους 64-bit. Μετά από κάθε απορροφόμενο ή συμπιεζόμενο μπλοκ (εκτός του πρώτου μπλοκ σύνοψης H_1), η συνάρτηση αντιμετάθεσης p^b είναι αυτή που εφαρμόζεται στην κατάσταση κάθε γύρου. Αντίθετα, η συνάρτηση αντιμετάθεσης p^a εφαρμόζεται

μόνο κατά την αρχικοποίηση και κατά την οριστικοποίηση μετά το τελευταίο μπλοκ μηνύματος M_s . Και τα δύο είδη συναρτήσεων, σύνοψης και επεκτάσιμης εξόδου, παρέχουν ασφάλεια 128-bit με μέγεθος σύνοψης τουλάχιστον 256-bit, και χρησιμοποιούν την ίδια ελαφρά αντιμετάθεση 320-bit με τις λειτουργίες πιστοποιημένης κρυπτογράφησης.



Σχήμα 13.5: Διαδικασία δημιουργίας σύνοψης στην κρυπτογραφική σουίτα ASCON.

Οι αριθμοί των γύρων a και b , το μέγεθος σύνοψης h , καθώς και ο ρυθμός r και η χωρητικότητα c της κατασκευής Sponge, εξαρτώνται από την παραλλαγή της συνάρτησης σύνοψης ASCON. Οι προτεινόμενες παράμετροι για τις συναρτήσεις σύνοψης ASCON συνοψίζονται στον Πίνακα 13.4.

Πίνακας 13.4: Παράμετροι για τις συναρτήσεις σύνοψης ASCON.

Αλγόριθμος	Μέγεθος (bits)			Γύροι	
	Σύνοψη	Ρυθμός	Χωρητικότητα	p^a	p^b
ASCON-HASH	256	64	256	12	12
ASCON-XOF	d (αυθαίρετο)	64	256	12	12
ASCON-HASHA	256	64	256	12	8
ASCON-XOFA	d (αυθαίρετο)	64	256	12	8

13.3.2 Αλγόριθμος GIFT-COFB

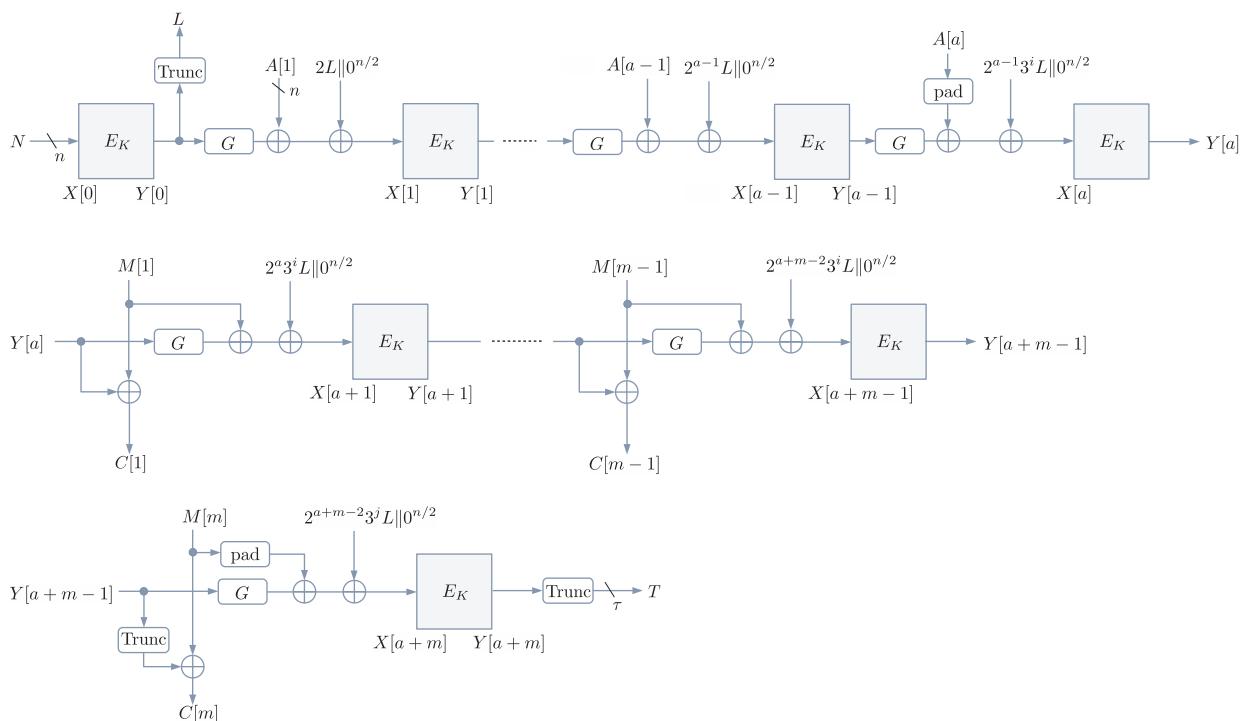
Η κρυπτογραφική σουίτα GIFT-COFB [6] αποτελεί ένα σύστημα κρυπτογράφησης που βασίζεται σε κρυπτογράφηση μπλοκ και χρησιμοποιεί συγκεκριμένα τον αλγόριθμο GIFT-128 [14]. Ουσιαστικά, ο αλγόριθμος GIFT-COFB μπορεί να θεωρηθεί ως μία αποτελεσματική ενσωμάτωση της λειτουργίας COFB [15] και της κρυπτογράφησης μπλοκ GIFT-128. Ο αλγόριθμος κρυπτογράφησης GIFT-128 διαθέτει μια εσωτερική κατάσταση 128-bit και επίσης ένα κλειδί 128-bit. Για την ακρίβεια, ο GIFT αποτελεί μία οικογένεια κρυπτογράφησης μπλοκ που παραμετροποιείται με βάση το μέγεθος κατάστασης και το μέγεθος κλειδιού και όλα τα μέλη αυτής της οικογένειας μπορούν να αναπτυχθούν αποτελεσματικά σε ελαφρές εφαρμογές. Από την άλλη πλευρά, η λειτουργία COFB υπολογίζει τη «συνδυασμένη ανατροφοδότηση» (COmbined FeedBack) της εξόδου κρυπτογράφησης και του μπλοκ δεδομένων με σκοπό την αύξηση του επιπέδου προστασίας. Αυτό στην πραγματικότητα βοηθά στο να σχεδιαστεί και να υλοποιηθεί ένα κρυπτογραφικό σχήμα με μια μικρού μεγέθους κατάσταση. Αυτή η τεχνική στην πραγματικότητα είναι ανθεκτική σε επιτιθέμενους που μπορούν να ελέγξουν το μπλοκ εισόδου και ταυτόχρονα την είσοδο κρυπτογράφησης του επόμενου μπλοκ. Συνολικά, ο συνδυασμός GIFT και COFB μπορεί να θεωρηθεί ως μία αρκετά αποτελεσματική, ελαφρά, και μικρού μέγεθος κατάστασης κρυπτογράφηση μπλοκ που βασίζεται στην κατασκευή AEAD.

Το σχήμα GIFT θεωρείται ένα από τα πιο ελαφρά κρυπτογραφικά σχήματα που υπάρχουν στη βιβλιογραφία. Αναφέρεται επίσης ως “Small PRESENT” καθώς ο σχεδιασμός του ακολουθεί αυτόν του κρυπτογραφικού σχήματος PRESENT [16]. Ωστόσο, ο GIFT είναι απαλλαγμένος από διάφορες γνωστές αδυναμίες που υπάρχουν στον PRESENT όσον αφορά τη γραμμική κρυπτανάλυση. Συνολικά, ο GIFT υπόσχεται αρκετά μεγαλύτερη απόδοση (τόσο ελαφρύτερη όσο και γρηγορότερη) σε σχέση με τον PRESENT. Ο αλγόριθμος GIFT διαθέτει έναν πολύ απλό σχεδιασμό που ξεπερνά σε απόδοση ακόμη και τους αλγορίθμους SIMON [17] και SKINNY [18]. Αποτελείται από πολύ απλές λειτουργίες, έτσι ώστε ο συνολικός χώρος που απαιτεί σε υλικό (hardware) σχεδόν να διατίθεται για την αποθήκευση του κρυπτοκειμένου. Η σχεδίασή του είναι σχεδόν βέλτιστη, καθώς εάν γινόταν χρήση ενός πιο αδύναμου πίνακα αντικαταστάσεων S-box (από αυτόν του GIFT) θα οδηγούσε σε ένα πιο αδύναμο κρυπτογραφικό σχήμα. Επιπλέον, ο σχεδιασμός του δεν βασίζεται στην υλοποίηση γύρων σε επίπεδό υλικού και οι διάφορες σταθερές που χρησιμοποιούνται παράγονται με την χρήση ενός πολύ ελαφρού καταχωρήτη LFSR (Linear-Feedback Shift Register). Τέλος, η δημιουργία των επιμέρους κλειδιών ανά γύρο είναι επίσης αρκετά ελαφριά και υλοποιείται απλώς από ολισθητές (shifts).

13.3.2.1 Λειτουργία Πιστοποιημένης Κρυπτογράφησης GIFT-COFB

Ο αλγόριθμος πιστοποιημένης κρυπτογράφησης, $GIFT\text{-}COFB}(K, N, A, M) \mapsto (C, T)$, λαμβάνει ως είσοδο ένα κλειδί κρυπτογράφησης $K \in \{0,1\}^{128}$, έναν μοναδικό αριθμό (nonce) $N \in \{0,1\}^{128}$, τα συσχετισμένα δεδομένα $A \in \{0,1\}^*$ και ένα μήνυμα $M \in \{0,1\}^*$. Ο μοναδικός αριθμός N μπορεί να υλοποιείται με την χρήση ενός μετρητή έτσι ώστε να διασφαλιστεί η μοναδικότητα του αριθμού αυτού. Ως έξοδο ο αλγόριθμος έχει ένα κρυπτοκειμένο $C \in \{0,1\}^{|M|}$ και μια ετικέτα $T \in \{0,1\}^{128}$. Η ακριβής διαδικασία κρυπτογράφησης απεικονίζεται στο Σχήμα 13.6.

Ο αλγόριθμος αποκρυπτογράφησης, $GIFT\text{-}COFB}^{-1}(K, N, A, C, T) \mapsto M$, λαμβάνει ως είσοδο τα (K, N, A, C, T) και έχει ως έξοδο το μήνυμα $M \in \{0,1\}^{|C|}$ εάν γίνει επαλήθευση της ετικέτας T ή το σύμβολο \perp σε διαφορετική περίπτωση.



Σχήμα 13.6: Διαδικασία κρυπτογράφησης στον αλγόριθμο GIFT-COFB.

13.3.2.2 Δομικά Στοιχεία του COFB

Κρυπτογράφηση Μπλοκ: Ο αλγόριθμος κρυπτογράφησης E_K αποτελείται από μια 128-bit κρυπτογράφηση μπλοκ με κλειδί 128-bit, όπως αυτή του GIFT-128 [14], αλλά με μια μικρή αλλαγή στη μορφή των δεδομένων εισόδου και εξόδου. Περισσότερες λεπτομέρειες σχετικά με αυτή την κρυπτογράφηση μπλοκ δίνονται στην επόμενη υποενότητα.

Συνάρτηση Πλήρωσης: Για $x \in \{0,1\}^*$ και δεδομένης μιας κενής συμβολοσειράς ϵ , η συνάρτηση πλήρωσης (padding) Pad ορίζεται ως εξής:

$$Pad(x) = \begin{cases} x & \text{εάν } x \neq \epsilon \text{ και } |x| \mod n = 0 \\ x \parallel 10^{(n-(|x| \mod n)-1)} & \text{σε διαφορετική περίπτωση} \end{cases}$$

Σημειώνεται ότι $Pad(\epsilon) = 10^{n-1}$.

Συνάρτηση Ανατροφοδότησης: Έστω ότι $Y \in \{0,1\}^{128}$ και $(Y[1], Y[2]) \xleftarrow{64} Y$, όπου $Y[i] \in \{0,1\}^{64}$. Οπότε μια συνάρτηση $G : \{0,1\}^{128} \rightarrow \{0,1\}^{128}$ ορίζεται ως εξής:

$$G(Y) = (Y[2], Y[1] \lll_1)$$

όπου για μια συμβολοσειρά X , το $X \lll_r$ υποδηλώνει μια αριστερή ολίσθηση του X κατά r bits. Επίσης, η συνάρτηση G μπορεί να οριστεί ως ένας 128×128 μη ιδιάζων (non-singular) δυαδικός πίνακας, επομένως η συνάρτηση $G(Y)$ μπορεί να εκφραστεί εναλλακτικά και ως $G \cdot Y$. Με δεδομένο ότι $M \in \{0,1\}^{128}$ και $Y \in \{0,1\}^{128}$, ορίζεται ότι $\rho_1(Y, M) = G \cdot Y \oplus M$. Η συνάρτηση ανατροφοδότησης (feedback) ρ και η αντίστοιχη της ρ' ορίζονται ως εξής:

$$\begin{aligned} \rho(Y, M) &= (\rho_1(Y, M), Y \oplus M) \\ \rho'(Y, C) &= (\rho_1(Y, Y \oplus C), Y \oplus C) \end{aligned}$$

Θα πρέπει να σημειωθεί ότι εάν $(X, M) = \rho'(Y, C)$ τότε $X = (G \oplus I) \cdot Y \oplus C$, όπου I αποτελεί έναν 128×128 πίνακα ταυτότητας. Η κατάλληλη επιλογή του G διασφαλίζει ότι το $G \oplus I$ έχει κατάταξη $n - 1$ (για την ακρίβεια 127, όταν η κατασκευή έχει $n = 128$). Όταν το Y επιλέγεται τυχαία, τόσο το $\rho_1(Y, M)$ (κατά την κρυπτογράφηση) όσο και το $\rho_1(Y, Y \oplus C)$ (κατά την αποκρυπτογράφηση) παρουσιάζουν επίσης σχεδόν πλήρη εντροπία (δηλ. τυχαιότητα).

13.3.2.3 Δομικά Στοιχεία του GIFT

Ο αλγόριθμός GIFT-128 αποτελεί μια 128-bit κρυπτογράφηση μπλοκ που βασίζεται σε ένα δίκτυο αντικαταστάσεων-αντιμεταθέσεων (Substitution-Permutation Network – SPN) με μήκος κλειδιού 128-bit. Επιπλέον, ο αλγόριθμος αυτός περιλαμβάνει μια επαναληπτική διαδικασία 40 γύρων με πανομοιότυπη λειτουργία. Αρχικά, ο GIFT-128 πραγματοποιεί αρχικοποίηση της εσωτερικής του κατάστασης και στην συνέχεια σε κάθε γύρο του εκτελούνται οι συναρτήσεις των βημάτων: SubCells, PermBits και AddRoundKey. Τέλος, πριν από την εκτέλεση της συνάρτησης του βήματος AddRoundKey, καλείται η συνάρτηση ενημέρωσης κατάστασης κλειδιού η οποία πραγματοποιεί κατάλληλες αλλαγές στο κλειδί που θα χρησιμοποιηθεί σε κάθε γύρο.

Αρχικοποίηση: Το απλό κείμενο P μεγέθους 128-bit φορτώνεται στην κατάσταση κρυπτογράφησης S που εκφράζεται ως 4 τμήματα των 32-bit, $S = \{S_0, S_1, S_2, S_3\}$, όπου $S_i \in \{0,1\}^{32}$. Από την άλλη πλευρά, το μυστικό κλειδί 128-bit K φορτώνεται στην κατάσταση κλειδιού KS που εκφράζεται ως 8 λέξεις των 16-bit, $KS =$

$\{W_0, W_1, \dots, W_7\}$, όπου $W_i \in \{0,1\}^{16}$.

$$S = \begin{bmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \end{bmatrix} \leftarrow \begin{bmatrix} B_0 & || & B_1 & || & B_2 & || & B_3 \\ B_4 & || & B_5 & || & B_6 & || & B_7 \\ B_8 & || & B_9 & || & B_{10} & || & B_{11} \\ B_{12} & || & B_{13} & || & B_{14} & || & B_{15} \end{bmatrix}$$

$$KS = \begin{bmatrix} W_0 & || & W_1 \\ W_2 & || & W_3 \\ W_4 & || & W_5 \\ W_6 & || & W_7 \end{bmatrix} \leftarrow \begin{bmatrix} B_0 \parallel B_1 & || & B_2 \parallel B_3 \\ B_4 \parallel B_5 & || & B_6 \parallel B_7 \\ B_8 \parallel B_9 & || & B_{10} \parallel B_{11} \\ B_{12} \parallel B_{13} & || & B_{14} \parallel B_{15} \end{bmatrix}$$

όπου B_i είναι τα bytes εισόδου στον αλγόριθμο κρυπτογράφησης.

Συνάρτηση SubCells: Η συνάρτηση SubCells περιλαμβάνει το ακόλουθο σύνολο εντολών:

$$\begin{aligned} S_1 &\leftarrow S_1 \oplus (S_0 \wedge S_2) \\ S_0 &\leftarrow S_0 \oplus (S_1 \wedge S_3) \\ S_2 &\leftarrow S_2 \oplus (S_0 \vee S_1) \\ S_3 &\leftarrow S_3 \oplus S_2 \\ S_1 &\leftarrow S_1 \oplus S_3 \\ S_3 &\leftarrow \neg S_3 \\ S_2 &\leftarrow S_2 \oplus (S_0 \wedge S_1) \\ \{S_0, S_1, S_2, S_3\} &\leftarrow \{S_3, S_1, S_2, S_0\} \end{aligned}$$

όπου \oplus, \wedge, \vee και \neg υποδηλώνουν τις πράξεις XOR, AND, OR και NOT, αντίστοιχα.

Συνάρτηση PermBits: Η αντιστοίχηση ενός τμήματος S_i της κατάστασης κρυπτογράφησης S σε 32 μεμονωμένα bit γίνεται ως εξής τρόπο:

$$(S_i[31], S_i[30], \dots, S_i[0]) \xleftarrow{1} S_i$$

οπότε η συνάρτηση PermBits ορίζεται ως:

$$PermBits(S) = \{Pb_0(S_0), Pb_1(S_1), Pb_2(S_2), Pb_3(S_3)\}$$

όπου το Pb_i καθορίζεται από έναν πίνακα αντιμετάθεσης (βλέπε Πίνακα 1 του [6]).

Συνάρτηση AddRoundKey: Η συνάρτηση AddRoundKey ορίζεται ως εξής:

$$AddRoundKey(S, KS, i) = \{S_0, S_1 \oplus (W_6 \parallel W_7), S_2 \oplus (W_2 \parallel W_3), S_3 \oplus Const_i\}$$

όπου $Const_i = 0x800000XY$ αποτελεί μια σταθερά του i -ου γύρου και το byte $XY = 00c_5c_4c_3c_2c_1c_0$ είναι μια σταθερά γύρου που δημιουργείται χρησιμοποιώντας έναν LFSR καταχωρήτη των 6-bit, του οποίου η κατάσταση ενημερώνεται ως εξής:

$$c_5 \parallel c_4 \parallel c_3 \parallel c_2 \parallel c_1 \parallel c_0 \leftarrow c_4 \parallel c_3 \parallel c_2 \parallel c_1 \parallel c_0 \parallel c_5 \oplus c_4 \oplus 1$$

Τα έξι bits, c_i , αρχικοποιούνται στο μηδέν και ενημερώνονται πριν χρησιμοποιηθούν στον επόμενο γύρο.

Συνάρτηση Ενημέρωσης Κατάστασης Κλειδιού: Η συνάρτηση ενημέρωσης κατάστασης κλειδιού, $KeyUpdate$, ορίζεται ως εξής:

$$KeyUpdate(KS) = \{W_6 \ggg_2, W_7 \ggg_{12}, W_0, W_1, W_2, W_3, W_4, W_5\}$$

13.3.3 Αλγόριθμος SPARKLE (SCHWAEMM και ESCH)

Η κρυπτογραφική σουίτα SPARKLE [11, 19] βασίζεται στην αντιμετάθεση και στην κατασκευή Sponge (Ενότητα 3.2.4), και παρέχει πιστοποιημένη κρυπτογράφηση με συσχετισμένα δεδομένα (AEAD) και λειτουργία συνάρτησης σύνοψης. Πιο συγκεκριμένα, στο σχήμα αυτό καθορίζονται το σχήμα πιστοποιημένης κρυπτογράφησης SCHWAEMM και η οικογένεια κρυπτογραφικών συναρτήσεων σύνοψης ESCH. Ο στόχος αυτών των αλγορίθμων είναι να χρησιμοποιούν όσο το δυνατόν λιγότερους κύκλους επεξεργαστή (CPU cycles) για να εκτελέσουν τις λειτουργίες τους, διατηρώντας παράλληλα ισχυρές εγγυήσεις ασφαλείας και μικρό μέγεθος υλοποίησης.

Η κύρια έκδοση του σχήματος κρυπτογράφησης SCHWAEMM είναι η SCHWAEMM256-128 που παίρνει ως είσοδο έναν μοναδικό αριθμό (nonce) 256-bit, ένα κλειδί 128-bit και έχει ως έξοδο μια ετικέτα πιστοποίησης 128-bit. Το επίπεδο ασφάλειας που επιτυγχάνει είναι της τάξης των 120-bit όσον αφορά την εμπιστευτικότητα και την ακεραιότητα. Επιπλέον, παρέχονται τρεις άλλες παραλλαγές του σχήματος SCHWAEMM, η SCHWAEMM128-128, η SCHWAEMM192-192 και η SCHWAEMM256-256, που διαφέρουν ως προς το μέγεθος του κλειδιού, του μοναδικού αριθμού (nonce) και της ετικέτας πιστοποίησης, καθώς και ως προς στο επίπεδο ασφάλειας που επιτυγχάνουν.

Η κύρια έκδοση του σχήματος σύνοψης ESCH είναι η Esch256 που παράγει μια σύνοψη 256-bit, προσφέροντας ένα επίπεδο ασφάλειας της τάξης των 128-bit όσον αφορά τις ακόλουθες ιδιότητες ασφαλείας: αντίσταση πρώτου ορίσματος, αντίσταση δεύτερου ορίσματος και δυσκολίας εύρεσης συγκρούσεων (βλέπε Ορισμό 3.1 στην Ενότητα 3.1). Το κύριο δομικό της στοιχείο είναι η οικογένεια αντιμεταθέσεων SPARKLE384. Επιπλέον, παρέχεται η παραλλαγή Esch384 που βασίζεται στην οικογένεια αντιμεταθέσεων SPARKLE512, έχοντας ως έξοδο μια σύνοψη 384-bit και παρέχοντας ένα επίπεδο ασφάλειας της τάξης των 192-bit. Και οι δύο αυτές εκδόσεις συναρτήσεων σύνοψης χρησιμεύουν επίσης ως βάση για δύο συναρτήσεις επεκτάσιμης έξόδου (Extendable-Output Function – XOF), την XOEsch256 και την XOEsch384.

13.3.3.1 Αντιμετάθεση SPARKLE

Τα σχήματα πιστοποιημένης κρυπτογράφησης SCHWAEMM και σύνοψης ESCH χρησιμοποιούν την οικογένεια αντιμεταθέσεων SPARKLE που περιγράφεται ακολούθως. Συγκεκριμένα, η οικογένεια SPARKLE αποτελείται από τις αντιμεταθέσεις SPARKLE 256_{n_s} , SPARKLE 384_{n_s} και SPARKLE 512_{n_s} , με μεγέθη μπλοκ: 256, 384 και 512 bit, αντίστοιχα (Πίνακας 13.5). Η παράμετρος n_s αναφέρεται στον αριθμό των βημάτων και μια αντιμετάθεση μπορεί να οριστεί για οποιαδήποτε $n_s \in \mathbb{N}$.

Η δομή υψηλού επιπέδου της αντιμετάθεσης SPARKLE παρουσιάζεται στο Σχήμα 13.7. Οι αντιμεταθέσεις που απεικονίζονται σε αυτό το σχήμα κατασκευάζονται χρησιμοποιώντας τα ακόλουθα δύο κύρια στοιχεία:

- Μια κρυπτογράφηση μπλοκ 64-bit και με κλειδί 32-bit που ονομάζεται ARX-box Alzette [20] (συμβολίζεται με A) και δίνεται ως εξής:

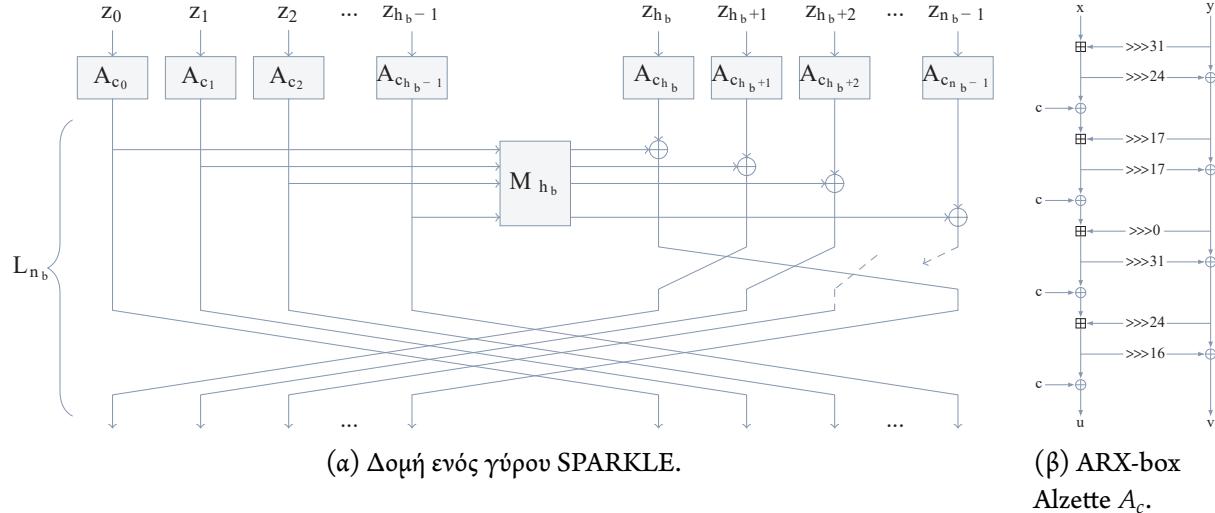
$$A : (\{0,1\}^{32} \times \{0,1\}^{32}) \times \{0,1\}^{32} \rightarrow (\{0,1\}^{32} \times \{0,1\}^{32}), ((x,y), c) \mapsto (u, v)$$

Ορίζεται το A_c ως μια αντιμετάθεση $(x, y) \mapsto A(x, y, c)$ από $\{0,1\}^{32} \times \{0,1\}^{32}$ σε $\{0,1\}^{32} \times \{0,1\}^{32}$.

- Ένα γραμμικό επίπεδο διάχυσης $L_{n_b} : \{0,1\}^{64n_b} \rightarrow \{0,1\}^{64n_b}$, όπου n_b υποδηλώνει τον αριθμό των διακλαδώσεων 64-bit (δηλ., το μέγεθος του μπλοκ διαιρούμενο με το 64) και είναι απαραίτητο να είναι άρτιο.

Πιο αναλυτικά, η δομή αντιμετάθεσης SPARKLE αποτελεί μια κλασική κατασκευή ενός δικτύου αντικαταστάσεων-αντιμεταθέσεων (Substitution-Permutation Network – SPN) εκτός από το ότι οι συναρτήσεις που παίζουν το ρόλο των S-box είναι διαφορετικές σε κάθε διακλάδωση. Κάθε μέλος της οικογένειας αντιμεταθέσεων επαναλαμβάνει μια παράλληλη εφαρμογή του Alzette κάτω από διαφορετικές, εξαρτώμενες από την διακλάδωση, σταθερές c_i (δείτε λεπτομέρειες στο Σχήμα 13.7β). Στην συνέχεια, ακολουθείται από την

εφαρμογή ενός γραμμικού επιπέδου διάχυσης L_{n_b} σε όλες τις διακλαδώσεις. Στην αντιμετάθεση SPARKLE, μια τέτοια παράλληλη εφαρμογή του Alzette που ακολουθείται από ένα γραμμικό επίπεδο διάχυσης, καλείται γύρος.



Σχήμα 13.7: Συνολική δομή ενός γύρου SPARKLE και του αντίστοιχου ARX-box Alzette A_c .

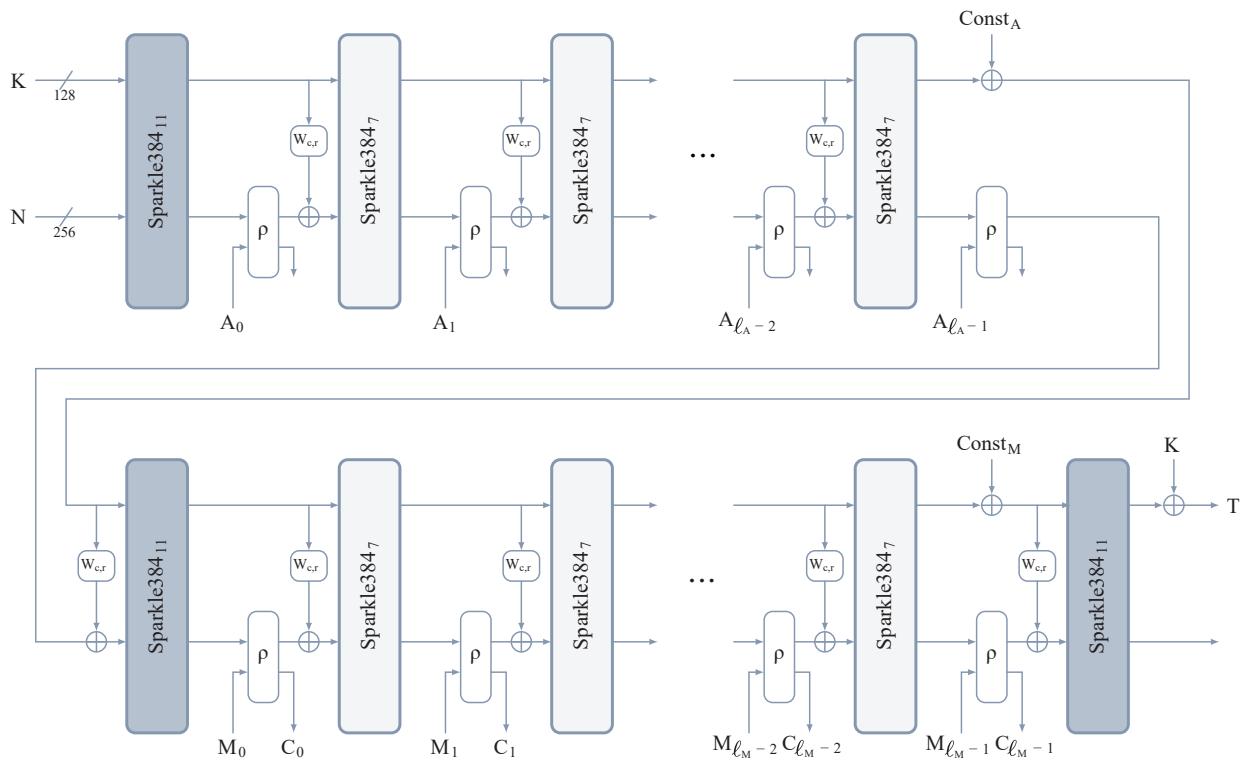
Πίνακας 13.5: Διαφορετικές εκδόσεις της αντιμετάθεσης SPARKLE.

Αντιμετάθεση	n	Αριθμός Βημάτων n_s	
		Ελαφρά Έκδοση	Πλήρης Έκδοση
SPARKLE256	256	7	10
SPARKLE384	384	7	11
SPARKLE512	512	8	12

13.3.3.2 Λειτουργίες Πιστοποιημένης Κρυπτογράφησης SCHWAEMM

Οι προτεινόμενες παραλλαγές πιστοποιημένης κρυπτογράφησης SCHWAEMM είναι η SCHWAEMM128-128, η SCHWAEMM256-128, η SCHWAEMM192-192 και η SCHWAEMM128-128, όπου για ένα δεδομένο κλειδί K και έναν μοναδικό αριθμό (nonce) N , επιτρέπουν την χρήση συσχετισμένων δεδομένων A και μηνυμάτων M αυθαίρετου μήκους και έχουν ως έξοδο ένα κρυπτοκείμενο C (με $|C| = |M|$) και μια ετικέτα πιστοποίησης T . Αντίστοιχα, από μια δεδομένη αποκρυπτογράφηση (K, N, A, C, T) μπορεί να ανακτηθεί το αρχικό μήνυμα M εάν η ετικέτα T είναι έγκυρη, διαφορετικά επιστρέφει το σύμβολο σφάλματος \perp . Το κύριο μέλος του σχήματος SCHWAEMM είναι το SCHWAEMM256-128 που παρουσιάζεται στο Σχήμα 13.8. Όλες οι παραλλαγές χρησιμοποιούν (με μικρές διαφοροποιήσεις) τον τρόπο λειτουργίας Beetle [21], ο οποίος βασίζεται στην κατασκευή Sponge Duplex [22].

Οι διαφορές μεταξύ των παραλλαγών του SCHWAEMM είναι η έκδοση της υποκείμενης αντιμετάθεσης SPARKLE (και επομένως η τιμή ρυθμού (rate) και χωρητικότητας (capacity)) και το μέγεθος της ετικέτας πιστοποίησης. Η ονοματοδοσία των παραλλαγών SCHWAEMM ακολουθεί την μορφή SCHWAEMMr-c, όπου το r υποδηλώνει το μέγεθος του ρυθμού και το c το μέγεθος της χωρητικότητας σε bits. Η πλήρης έκδοση της αντιμετάθεσης SPARKLE, όπως φαίνεται και στο Σχήμα 13.8, χρησιμοποιείται κατά την αρχικοποίηση, τον διαχωρισμό μεταξύ της επεξεργασίας των συσχετισμένων δεδομένων και του μηνύματος, καθώς και κατά



Σχήμα 13.8: Διαδικασία κρυπτογράφησης του κρυπτογραφικού σχήματος SCHWAEMM256-128.

την οριστικοποίηση, ενώ εναλλακτικά η ελαφρά έκδοση χρησιμοποιείται κατά την ενημέρωση των ενδιάμεσων καταστάσεων. Ο Πίνακας 13.6 παρέχει μια επισκόπηση των παραμέτρων στις διάφορες παραλλαγές του SCHWAEMM.

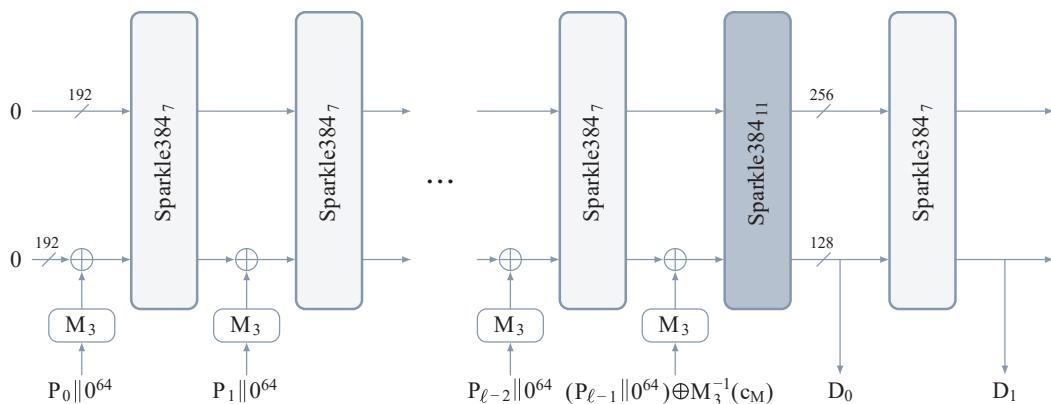
Πίνακας 13.6: Παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης SCHWAEMM.

Αλγόριθμος	Μέγεθος (bits)							Επίπεδο Ασφαλείας
	Κλειδί	Nonce	Ετικέτα	Εσωτ. Κατάσταση	Ρυθμός	Χωρητικότητα		
SCHWAEMM256-128	128	256	128	384	256	128		120-bit
SCHWAEMM192-192	192	192	192	384	192	192		184-bit
SCHWAEMM128-128	128	128	128	256	128	128		120-bit
SCHWAEMM256-256	256	256	256	512	256	256		248-bit

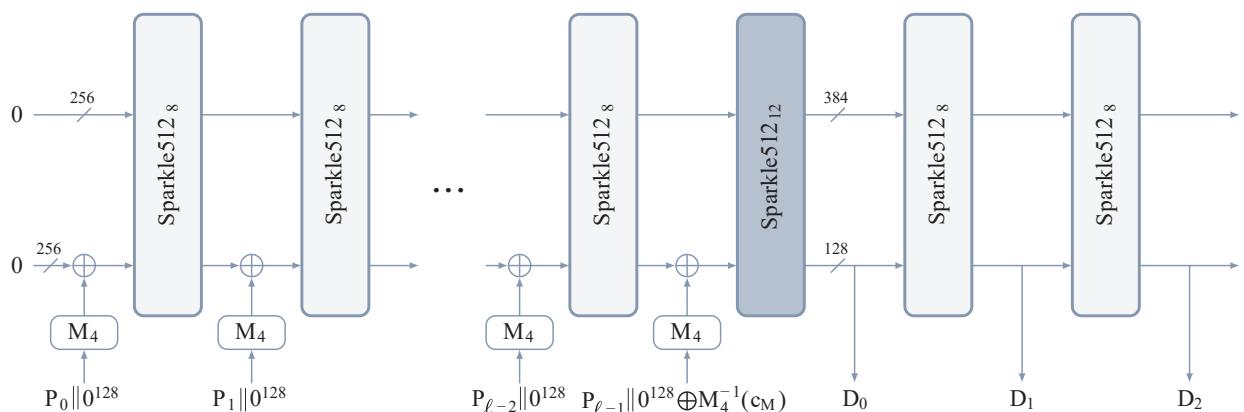
13.3.3.3 Λειτουργίες Συναρτήσεων Σύνοψης ESCH

Οι προτεινόμενες παραλλαγές των συναρτήσεων σύνοψης ESCH είναι η Esch256 και η Esch384, που επιτρέπουν την επεξεργασία ενός μηνύματος $M \in \{0,1\}^*$ αυθαίρετου μήκους και έχουν ως έξοδο μια σύνοψη D μεγέθους 256-bit και 384-bit, αντίστοιχα. Και οι δύο αυτές παραλλαγές χρησιμοποιούν τη γνωστή κατασκευή Sponge, η οποία κατασκευάζεται από αντιμεταθέσεις SPARKLE, και παραμετροποιείται από τον ρυθμό r και την χωρητικότητα c . Η ελαφρά έκδοση της αντιμετάθεσης SPARKLE χρησιμοποιείται τόσο κατά την απορρόφηση (absorption) όσο και κατά τη συμπίεση (squeezing), ενώ η πλήρης έκδοση χρησιμοποιείται μόνο μεταξύ των δύο αυτών φάσεων (βλέπε Σχήμα 13.9). Τόσο η Esch256 όσο και η Esch384, χρησιμοποιούν τον ίδιο προκαθορισμένο ρυθμό r των 128-bit. Αυτό σημαίνει ότι η πλήρωση (padding) του μηνύματος M πρέπει να γίνει με τέτοιο τρόπο ώστε το μήκος του μηνύματος σε bit να αποτελεί ακέραιο πολλαπλά-

σιο του 128. Για τον σκοπό αυτό, ο κανόνας πλήρωσης που ακολουθείται είναι η απλή προσθήκη του 10^* . Τα Σχήματα 13.9α και 13.9β παρουσιάζουν την διαδικασία δημιουργίας σύνοψης για την Esch256 και την Esch384, αντίστοιχα. Σύμφωνα με τα σχήματα, τα μπλοκ μηνυμάτων (64 ή 128 bits) εισάγονται έμμεσα, δηλαδή, πρώτα γίνεται η πλήρωση τους, ακολουθεί ο μετασχηματισμός τους μέσω των συναρτήσεων M_3 και M_4 για την σύνοψη Esch256 και Esch384, αντίστοιχα, και τέλος η ακολουθία από bits που προκύπτει γίνεται XOR με τα αριστερότερα bits της εσωτερικής κατάστασης.



(a) Συνάρτηση σύνοψης Esch256.



(b) Συνάρτηση σύνοψης Esch384.

Σχήμα 13.9: Διαδικασία δημιουργίας σύνοψης του κρυπτογραφικού σχήματος ESCH.

Επιπλέον, οι συναρτήσεις σύνοψης Esch256 και Esch384 μπορούν εύκολα να προσαρμοστούν έτσι ώστε να παρέχουν εξόδους αυθαίρετου μήκους. Έτσι, προτείνονται οι συναρτήσεις επεκτάσιμης εξόδου (XOF) XOEsch256 και XOEsch384, οι οποίες είναι αρκετά παρόμοιες με τις αντίστοιχες συναρτήσεις σύνοψης. Εκτός από το ότι χρησιμοποιούν άλλες τιμές για τις σταθερές C_M , η κύρια διαφορά τους είναι ότι οι παραλλαγές XOF διαθέτουν ως πρόσθετη παράμετρο εισόδου το t που καθορίζει το μέγεθος της σύνοψης εξόδου. Σε αυτή την περίπτωση, η φάση συμπίεσης (squeezing) επεκτείνεται κατάλληλα έτσι ώστε να παρέχει την έξοδο του απαιτούμενου μεγέθους. Ο Πίνακας 13.7 παρέχει μια επισκόπηση των παραμέτρων στις διάφορες παραλλαγές του ESCH.

13.3.4 Αλγόριθμος TinyJAMBU

Η κρυπτογραφική σουίτα TinyJAMBU [12] αποτελεί μια οικογένεια αλγορίθμων ελαφράς πιστοποιημένης κρυπτογράφησης που υιοθετεί μια μικρή παραλλαγή του σχήματος JAMBU [23]. Η τελευταία αποτελεί την μικρότερη λειτουργία πιστοποιημένης κρυπτογράφησης που υποβλήθηκε στον διαγωνισμό CAESAR (2014-2019) και επιλέχθηκε στον τρίτο γύρο του διαγωνισμού. Η αρχή λειτουργίας του TinyJAMBU βασίζεται σε

Πίνακας 13.7: Παράμετροι για τις συναρτήσεις σύνοψης ESCH.

Αλγόριθμος	Μέγεθος (bits)				Επίπεδο Ασφαλείας
	Σύνοψη	Εσωτ. Κατάστ.	Ρυθμός	Χωρητικότητα	
ESCH256	256	384	128	256	128-bit
ESCH384	384	512	128	384	192-bit
XOEsCH256	d (ανθαίρετο)	384	128	256	min{128, t}-bit
XOEsCH384	d (ανθαίρετο)	512	128	384	min{192, t}-bit

μια αντιμετάθεση με χρήση κλειδιού. Το μέγεθος της εσωτερικής κατάστασης του TinyJAMBU είναι μόνο τα δύο τρίτα ($2n$ -bit) του JAMBU, και το μέγεθος του μπλοκ μηνύματος είναι μόλις το μισό (n -bit) από αυτό του JAMBU. Στην περίπτωση που ο μοναδικός αριθμός nonce επαναχρησιμοποιηθεί, το σχήμα TinyJAMBU παρέχει καλύτερη ασφάλεια ελέγχου ταυτότητας από το σχήμα JAMBU. Επιπλέον, η ασφάλεια ελέγχου ταυτότητας του TinyJAMBU είναι επίσης καλύτερη από αυτή της κατασκευής Sponge Duplex [22] όταν ο μοναδικός αριθμός nonce επαναχρησιμοποιηθεί (για το ίδιο μέγεθος αντιμετάθεσης και το ίδιο μέγεθος μπλοκ μηνύματος).

13.3.4.1 Αντιμετάθεση P_n με Χρήση Κλειδιού

Στο σχήμα TinyJAMBU χρησιμοποιείται μια αντιμετάθεση 128-bit, P_n , με χρήση κλειδιού μεγέθους $klen$. Η αντιμετάθεση P_n αποτελείται συνολικά από n γύρους. Στον i -οστό γύρο της αντιμετάθεσης, χρησιμοποιείται ένας μη γραμμικός καταχωρητής NFSR 128-bit (Nonlinear Feedback Shift Register – NFSR) για την ενημέρωση της εσωτερικής κατάστασης και η οποία γίνεται ως εξής (βλέπε Σχήμα 13.10):

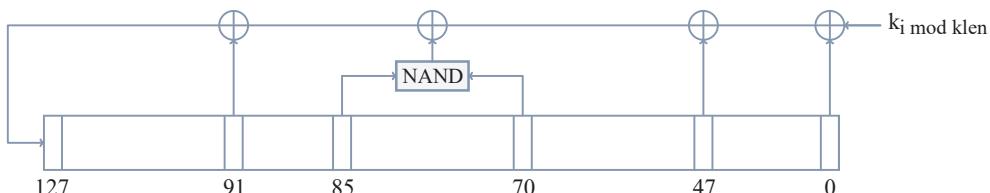
```

StateUpdate( $S, K, i$ )
   $feedback = s_0 \oplus s_{47} \oplus (\neg(s_{70} \wedge s_{85})) \oplus s_{91} \oplus k_i \bmod klen$ 
  for  $j = 0, \dots, 126$  do
     $s_j = s_{j+1}$ 
     $s_{127} = feedback$ 
  end

```

όπου \oplus , \wedge και \neg υποδηλώνουν τις πράξεις XOR, AND και NOT, αντίστοιχα.

Για παράδειγμα, η αντιμετάθεση P_{640} υποδηλώνει ότι η κατάσταση της αντιμετάθεσης ενημερώνεται χρησιμοποιώντας τη συνάρτηση $StateUpdate()$ για 640 φορές. Σε μια 32-bit αρχιτεκτονική επεξεργαστών μπορούν να εκτελεστούν παράλληλα 32 γύροι αντιμεταθέσεων σε έναν κύκλο.

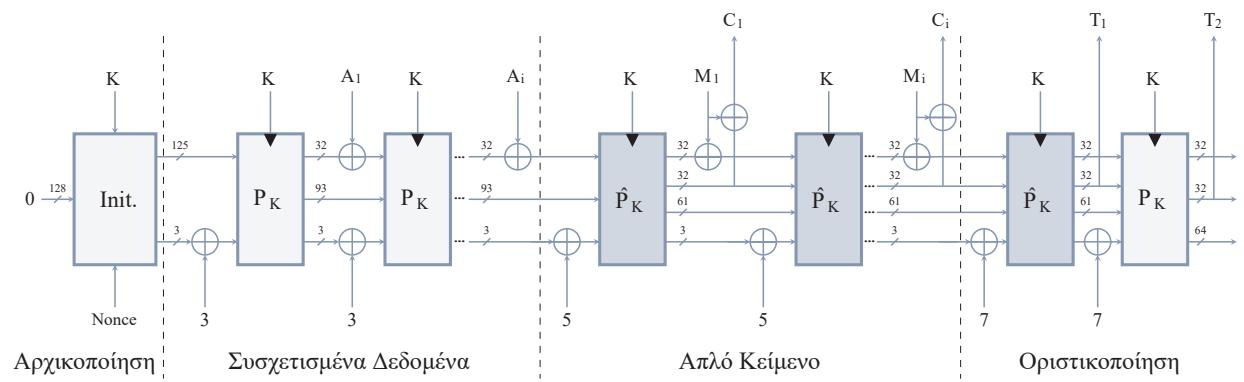


Σχήμα 13.10: Μη γραμμικός καταχωρητής NFSR 128-bit στο κρυπτογραφικό σχήμα TinyJAMBU.

13.3.4.2 Λειτουργίες Πιστοποιημένης Κρυπτογράφησης TinyJAMBU

Στη λειτουργία πιστοποιημένης κρυπτογράφησης του σχήματος TinyJAMBU, χρησιμοποιείται μια αντιμετάθεση 128-bit με χρήση κλειδιού, το μέγεθος εσωτερικής κατάστασης είναι 128-bit, το μέγεθος του μπλοκ μηνύματος είναι 32-bit και η ετικέτα πιστοποίησης είναι 64-bit. Η αντιμετάθεση με χρήση κλειδιού υποστηρίζει τρία πιθανά μεγέθη κλειδιών: 128, 192, και 256 bits. Όταν ο μοναδικός αριθμός nonce επαναχρησιμοποιηθεί, το σχήμα TinyJAMBU παρέχει καλύτερη ασφάλεια ελέγχου ταυτότητας από το σχήμα JAMBU και τη κατασκευή Sponge Duplex (για το ίδιο μέγεθος αντιμετάθεσης και το ίδιο μέγεθος μπλοκ μηνύματος). Ο λόγος είναι ότι ένας επιτιθέμενος μπορεί εύκολα να θέσει ένα μέρος της κατάστασης σε μια αυθαίρετη τιμή της επιλογής του όταν το nonce επαναχρησιμοποιηθεί στη κατασκευή Sponge Duplex, ενώ είναι δύσκολο να το κάνει αυτό στο σχήμα TinyJAMBU.

Η διαδικασία κρυπτογράφησης TinyJAMBU παρουσιάζεται στο Σχήμα 13.11. Στην περίπτωση που το τελευταίο μπλοκ των συσχετισμένων δεδομένων (ή του απλού κειμένου) δεν είναι πλήρες μπλοκ, τότε το μήκος του μη-πλήρους μπλοκ (σε bytes) γίνεται XOR με την εσωτερική του κατάσταση.



Σχήμα 13.11: Διαδικασία κρυπτογράφησης του κρυπτογραφικού σχήματος TinyJAMBU.

Οι διάφορες διεργασίες που πραγματοποιούνται στην πιστοποιημένη κρυπτογράφηση του σχήματος TinyJAMBU είναι οι εξής:

- **Αρχικοποίηση:** Η αρχικοποίηση του TinyJAMBU αποτελείται από τα ακόλουθα δύο στάδια:
 - **Εγκατάσταση κλειδιού.** Η εγκατάσταση κλειδιού πραγματοποιείται με την τυχαιοποίηση της κατάστασης $S \in \{0,1\}^{128}$, η οποία αρχικοποιείται σε 0, χρησιμοποιώντας την αντιμετάθεση $\hat{P}_K = P_{n=\{1024,1152,1280\}}$.
 - **Εγκατάσταση μοναδικού αριθμού nonce 96-bit.** Η εγκατάσταση μοναδικού αριθμού nonce πραγματοποιείται σε τρία βήματα. Σε κάθε βήμα, (1) τα bits πλαισίου (framebits) του nonce (με τιμή 1) γίνονται XOR με την κατάσταση, (2) ενημερώνεται η κατάσταση χρησιμοποιώντας την αντιμετάθεση $P_K = P_{640}$, και (3) 32 bits του nonce γίνονται XOR με την κατάσταση.
- **Επεξεργασία των συσχετισμένων δεδομένων:** Μετά την αρχικοποίηση γίνεται επεξεργασία των συσχετισμένων δεδομένων σε βήματα, ο αριθμός των οποίων εξαρτάται από το μέγεθος των συσχετισμένων δεδομένων χωρισμένων σε μπλοκ των 32-bit. Σε κάθε βήμα, (1) τα bits πλαισίου (framebits) των συσχετισμένων δεδομένων (με τιμή 3) γίνονται XOR με την εσωτερική κατάσταση, (2) ενημερώνεται η κατάσταση χρησιμοποιώντας την αντιμετάθεση $P_K = P_{640}$ και, (3) 32 bits των συσχετισμένων δεδομένων γίνονται XOR με την κατάσταση.
- **Κρυπτογράφηση:** Μετά την επεξεργασία των συσχετισμένων δεδομένων, πραγματοποιείται η κρυπτογράφηση του απλού κειμένου M σε βήματα, ο αριθμός των οποίων εξαρτάται από το μέγεθος του

απλού κειμένου χωρισμένου σε μπλοκ των 32-bit. Σε κάθε βήμα, (1) τα bits πλαισίου (framebits) του απλού κειμένου (με τιμή 5) γίνονται XOR με την κατάσταση, (2) ενημερώνεται η κατάσταση χρησιμοποιώντας την αντιμετάθεση $\hat{P}_K = P_{n=\{1024,1152,1280\}}$, (3) 32 bits του απλού κειμένου γίνονται XOR με την κατάσταση, και (4) εξάγονται 32 bits του κρυπτοκειμένου κάνοντας XOR το απλό κείμενο με ένα άλλο μέρος 32-bit της κατάστασης.

- Οριστικοποίηση:** Μετά την κρυπτογράφηση του απλού κειμένου, δημιουργείται η ετικέτα πιστοποίησης 64-bit T σε δύο βήματα. Σε κάθε βήμα, (1) τα bits πλαισίου (framebits) της οριστικοποίησης (με τιμή 7) γίνονται XOR με την κατάσταση, (2) ενημερώνεται η κατάσταση χρησιμοποιώντας στο πρώτο βήμα την αντιμετάθεση $\hat{P}_K = P_{n=\{1024,1152,1280\}}$ και στο δεύτερο την αντιμετάθεση $P_K = P_{640}$, και (3) εξάγονται 32 bits της ετικέτας ανά βήμα από ένα μέρος της κατάστασης.
- Αποκρυπτογράφηση:** Στην διαδικασία της αποκρυπτογράφησης, η αρχικοποίηση και η επεξεργασία των συσχετισμένων δεδομένων είναι ίδια με τη διαδικασία της κρυπτογράφησης. Μετά την επεξεργασία των συσχετισμένων δεδομένων, πραγματοποιείται η αποκρυπτογράφηση του κρυπτοκειμένου C σε βήματα, ο αριθμός των οποίων εξαρτάται από το μέγεθος του κρυπτοκειμένου χωρισμένου σε μπλοκ των 32-bit. Σε κάθε βήμα, (1) τα bits πλαισίου (framebits) του κρυπτοκειμένου (με τιμή 5) γίνονται XOR με την κατάσταση, (2) ενημερώνεται η κατάσταση χρησιμοποιώντας την αντιμετάθεση $\hat{P}_K = P_{n=\{1024,1152,1280\}}$, (3) 32-bit απλού κειμένου εξάγονται κάνοντας XOR του κρυπτοκειμένου με τα 32 bits κατάστασης $s_{\{64..95\}}$, και (4) το απλό κείμενο γίνεται XOR με την κατάσταση $s_{\{96..127\}}$.
- Επαλήθευση:** Μετά την αποκρυπτογράφηση του κρυπτοκειμένου, δημιουργείται η ετικέτα πιστοποίησης 64-bit T' σε δύο βήματα και, στη συνέχεια συγκρίνουμε την ετικέτα T' με την ετικέτα T που εξήχθη από την κρυπτογράφηση. Για την εξαγωγή της ετικέτας T' ακολουθείται η ίδια διαδικασία που αναφέρεται στην οριστικοποίηση.

Ο Πίνακας 13.8 παρέχει μια επισκόπηση των παραμέτρων για τις διάφορες παραλλαγές του TinyJAMBU κάνοντας αναφορά στις διαφοροποιήσεις που υπάρχουν όσον αφορά την αντιμετάθεση με χρήση κλειδιού $\hat{P}_K = P_{n=\{1024,1152,1280\}}$.

Πίνακας 13.8: Παράμετροι για τις λειτουργίες πιστοποιημένης κρυπτογράφησης TinyJAMBU.

Αλγόριθμος	P_n	Μέγεθος (bits)					Επίπεδο Ασφαλείας	
		Κλειδί	Nonce	Ετικέτα	Εσωτ. Κατάσταση	Κρυπτογραφ.	Αυθεντικ.	
TinyJAMBU-128	P_{1024}	128	96	64	128	112-bit	64-bit	
TinyJAMBU-192	P_{1152}	192	96	64	128	168-bit	64-bit	
TinyJAMBU-256	P_{1280}	256	96	64	128	224-bit	64-bit	

13.3.5 Αλγόριθμος Xoodyak

Η κρυπτογραφική σουίτα XOOODYAK [13] αποτελεί ένα ευέλικτο κρυπτογραφικό σχήμα που είναι κατάλληλο για τις περισσότερες λειτουργίες συμμετρικού κλειδιού, όπως συναρτήσεις σύνοψης, δημιουργία ψευδοτυχαίων bits, αυθεντικοποίηση, κρυπτογράφηση και πιστοποιημένη κρυπτογράφηση. Βασίζεται στην κατασκευή Sponge Duplex [22] και συγκεκριμένα σε μια παραλλαγή πλήρους κατάστασης [24] όταν τροφοδοτείται με ένα μυστικό κλειδί. Επίσης, μοιράζεται ανάλογα χαρακτηριστικά με τα ακόλουθα πρωτόκολλα: Blinker [25], STROBE [26] και SHO (Stateful Hash Objects) [27]. Στην πράξη, το σχήμα XOOODYAK είναι αρκετά απλό στη χρήση και η υλοποίηση του μπορεί να βρει χρήση σε πολλές διαφορετικές περιπτώσεις χρήσης.

Εσωτερικά, το σχήμα XOOODYAK χρησιμοποιεί την αντιμετάθεση XOODOO [28]. Η σχεδιαστική αυτή πρόσεγγιση αντιμετάθεσης 384-bit είναι εμπνευσμένη από το Keccak-p [29, 30], ενώ διαθέτει περισσότερες από μια διαστάσεις, όπως το Gimli [31], για καλύτερη αποδοτικότητα σε επεξεργαστές χαμηλών επιδόσεων. Η κατάστασή του αποτελείται από τρία ανεξάρτητα επίπεδα των 128-bit το καθένα, και τα οποία αλληλεπιδρούν ανά στήλες 3-bit μέσω μίξης και μη γραμμικών λειτουργιών. Η συνάρτηση γύρων που διαθέτει παρέχεται για επεξεργαστές χαμηλών επιδόσεων 32-bit, καθώς και για εξειδικευμένο υλικό.

Ο τρόπος λειτουργίας που εφαρμόζεται ως συνέχεια της αντιμετάθεσης XOODOO καλείται *Cyclist*, ως μια ελαφρά έκδοση του αντίστοιχου τρόπου λειτουργίας *Motorist* που χρησιμοποιείται στο KEYAK [32]. Ο τρόπος λειτουργίας *Cyclist* είναι αρκετά πιο απλός από το *Motorist*, κυρίως χάρη στην απουσία παράλληλων παραλλαγών. Μια άλλη σημαντική διαφορά είναι ότι το *Cyclist* δεν περιορίζεται μόνο στην πιστοποιημένη κρυπτογράφηση και υποστηρίζει επίσης συναρτήσεις σύνοψης.

Συνολικά, η κρυπτογραφική σουίτα XOOODYAK μπορεί να εξασφαλίσει επίπεδο ασφαλείας 128-bit τόσο σε λειτουργίες σύνοψης όσο και σε λειτουργίες πιστοποιημένης κρυπτογράφησης (υποθέτοντας ότι το κλειδί $k \geq 128$ bits).

13.3.5.1 Αντιμετάθεση Xoodoo

Η αντιμετάθεση XOODOO [28] αποτελεί μια οικογένεια αντιμεταθέσεων που παραμετροποιείται με βάση τον αριθμό των γύρων n_r , και συμβολίζονται ως $XOODOO[n_r]$. Η αντιμετάθεση XOODOO διαθέτει μια κλασική επαναλαμβανόμενη δομή με επαναληπτική εφαρμογή μιας συνάρτησης γύρων σε μια κατάσταση. Η κατάσταση αποτελείται από τρία ισομεγέθη οριζόντια επίπεδα, το καθένα από τα οποία αποτελείται από 4 παράλληλες λωρίδες των 32-bit. Ομοίως, η κατάσταση μπορεί να θεωρηθεί ως ένα σύνολο 128 στηλών των 3-bit, διατεταγμένων σε έναν πίνακα 4×32 . Τα επίπεδα δεικτοδοτούνται (indexed) με βάση το y , με το επίπεδο $y = 0$ στο κατώτερο επίπεδο και επίπεδο $y = 2$ στο ανώτερο επίπεδο. Μέσα σε μια λωρίδα, δεικτοδοτούνται τα bits με βάση το z . Οι λωρίδες μέσα σε ένα επίπεδο δεικτοδοτούνται με βάση το x , επομένως η θέση μιας λωρίδας στην κατάσταση καθορίζεται από τις δύο συντεταγμένες (x, y) . Τα bit της κατάστασης δεικτοδοτούνται με (x, y, z) και οι στήλες με (x, z) . Τα φύλλα είναι οι συστοιχίες τριών λωρίδων, η μια πάνω στην άλλη, και δεικτοδοτούνται με βάση το x . Η εσωτερική κατάσταση της αντιμετάθεσης XOODOO παρουσιάζεται στο Σχήμα 13.12.

Η αντιμετάθεση αποτελείται από την επανάληψη μιας συνάρτησης γύρων R_i που διαθέτει 5 βήματα: ένα στρώμα μίξης θ , ένα επίπεδο ολίσθησης ρ_{west} , μια προσθήκη σταθερών του γύρου i , ένα μη-γραμμικό στρώμα χ , και ένα επιπλέον επίπεδο ολίσθησης ρ_{east} . Οι πράξεις που πραγματοποιούνται σε αυτά τα βήματα αποτύπωνονται ως εξής:

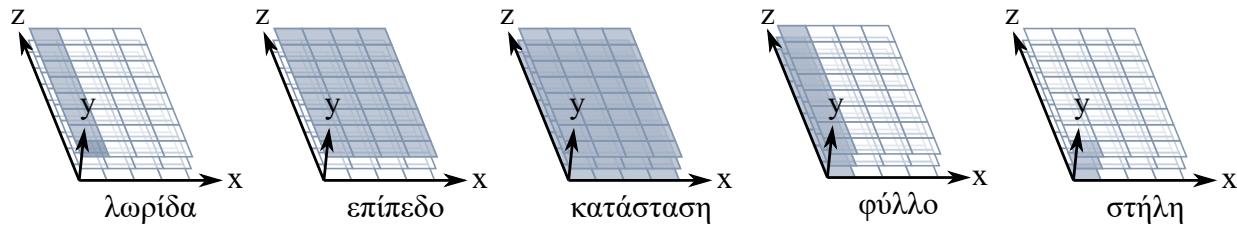
$$\begin{aligned} \text{Βήμα } \theta: \quad & P[x] = A[x, 0] \oplus A[x, 1] \oplus A[x, 2] & 0 \leq x \leq 3 \\ & E[x] = (P[x - 1] \lll_5) \oplus (P[x - 1] \lll_{14}) & 0 \leq x \leq 3 \\ & A[x, y] = A[x, y] \oplus E[x] & 0 \leq x \leq 3, 0 \leq y \leq 2 \end{aligned}$$

$$\begin{aligned} \text{Βήμα } \rho_{west}: \quad & A[x, 1] = A[x - 1, 1] & 0 \leq x \leq 3 \\ & A[x, 2] = A[x, 2] \lll_{11} & 0 \leq x \leq 3 \end{aligned}$$

$$\text{Βήμα } i: \quad A[0, 0] = A[0, 0] \oplus C_i$$

$$\text{Βήμα } \chi: \quad A[x, y] = A[x, y] \oplus ((\neg A[x, y + 1]) \wedge A[x, y + 2]) \quad 0 \leq x \leq 3, 0 \leq y \leq 2$$

$$\begin{aligned} \text{Βήμα } \rho_{east}: \quad & A[x, 1] = A[x, 1] \lll_1 & 0 \leq x \leq 3 \\ & A[x, 2] = A[x - 2, 2] \lll_8 & 0 \leq x \leq 3 \end{aligned}$$



Σχήμα 13.12: Εσωτερική κατάσταση της αντιμετάθεσης Xoodoo (για λόγους επίδειξης οι λωρίδες έχουν μειωθεί από 32 σε 8 bits).

Τέλος, σε πολλές εφαρμογές η κατάσταση της αντιμετάθεσης Xoodoo μπορεί να προσδιορίζεται ως μια συμβολοσειρά s 384-bit, και τα bits να δεικτοδοτούνται με βάση το i . Η αντιστοίχιση από την τρισδιάστατη δεικτοδότηση (x, y, z) στην μονοδιάστατη i γίνεται με βάση την εξίσωση $i = z + 32(x + 4y)$.

13.3.5.2 Τρόπος Λειτουργίας *Cyclist*

Ο τρόπος λειτουργίας *Cyclist* βασίζεται σε μια κρυπτογραφική αντιμετάθεση και παράγει ένα αντικείμενο κατάστασης στο οποίο ο χρήστης μπορεί να πραγματοποιεί κλήσεις. Παραμετροποιείται από τη αντιμετάθεση f , από τα μεγέθη μπλοκ R_{hash} , R_{kin} και R_{kout} , και από το μέγεθος αναστολέα (ratchet) $\ell_{ratchet}$ (όλα σε bytes). Τα μεγέθη μπλοκ R_{hash} , R_{kin} και R_{kout} καθορίζουν το μέγεθος σύνοψης, το μέγεθος εισόδου και το μέγεθος εξόδου σε λειτουργίες χρήστης κλειδιού, αντίστοιχα. Καθώς το *Cyclist* χρησιμοποιεί έως και 2 bytes για bits πλαισίου (δηλ., bits που χρησιμοποιούνται για πλήρωση (padding) και διαχωρισμό τομέα), απαιτείται ότι $\max(R_{hash}, R_{kin}, R_{kout}) + 2 \leq b'$, όπου $b' = b/8$ είναι το μέγεθος της αντιμετάθεσης σε bytes.

Κατά την αρχικοποίηση με $\text{CYCLIST}(K, id, counter)$, το αντικείμενο *Cyclist* μπορεί να ξεκινήσει είτε σε λειτουργία σύνοψης εάν $K = \epsilon$, είτε σε λειτουργία χρήστης κλειδιού σε διαφορετική περίπτωση. Στη δεύτερη περίπτωση, το αντικείμενο δέχεται το συμμετρικό κλειδί K μαζί με ένα (προαιρετικό) αναγνωριστικό id , και στη συνέχεια δέχεται ως είσοδο έναν μετρητή $counter$ με σταδιακό τρόπο εάν $counter \neq \epsilon$. Στην πρώτη περίπτωση, απλά αγνοεί τις παραμέτρους αρχικοποίησης. Επιπλέον, θα πρέπει να ληφθεί υπόψη, ότι σε αντίθεση με το STROBE [26], δεν υπάρχει τρόπος να μεταβεί το *Cyclist* από την λειτουργία σύνοψης στην λειτουργία χρήστης κλειδιού.

Οι διαθέσιμες συναρτήσεις που υποστηρίζονται εξαρτώνται από τον τρόπο εκκίνησης του αντικειμένου *Cyclist*: Οι συναρτήσεις $\text{ABSORB}()$ και $\text{SQUEEZE}()$ μπορούν να κληθούν τόσο σε λειτουργία σύνοψης όσο και σε λειτουργία χρήστης κλειδιού, ενώ οι συναρτήσεις $\text{ENCRYPT}()$, $\text{DECRYPT}()$, $\text{SQUEEZEKEY}()$ και $\text{RATCHET}()$ περιορίζονται στη λειτουργία χρήστης κλειδιού. Ο σκοπός κάθε συνάρτησης είναι ο εξής:

- Η $\text{ABSORB}(X)$ δέχεται ως είσοδο μια συμβολοσειρά X .
- Η $C \leftarrow \text{ENCRYPT}(P)$ κρυπτογραφεί το P σε C και απορροφά το P .
- Η $P \leftarrow \text{DECRYPT}(C)$ αποκρυπτογραφεί το C σε P και απορροφά το P .
- Η $Y \leftarrow \text{SQUEEZE}(\ell)$ παράγει μια έξοδο ℓ -byte που εξαρτάται από τα δεδομένα που έχουν απορροφηθεί μέχρι στιγμής.
- Η $Y \leftarrow \text{SQUEEZEKEY}(\ell)$ λειτουργεί όπως η $Y \leftarrow \text{SQUEEZE}(\ell)$ αλλά με στόχο τη δημιουργία ενός παράγωγου κλειδιού.
- Η $\text{RATCHET}()$ μετασχηματίζει την κατάσταση με έναν μη αναστρέψιμο τρόπο για να διασφαλίσει το απόρρητο.

13.3.5.3 Λειτουργία Πιστοποιημένης Κρυπτογράφησης *Xoodyak*

Η χρήση του σχήματος *XOODYAK* για πιστοποιημένη κρυπτογράφηση με συσχετισμένα δεδομένα (AEAD) μπορεί να γίνει με την ακόλουθη διαδικασία. Από προεπιλογή, η αντιμετάθεση που χρησιμοποιείται είναι η *Xoodoo* [28] με μέγεθος 48 bytes (ή $b = 384$ bits), το μέγεθος του κλειδιού είναι $k = 128$ bits. Δεν υπάρχει καθολικό αναγνωριστικό κλειδιού αλλά ένας μοναδικός αριθμός *nonce* (με μέγεθος 128 bits) που ενσωματώνεται στην παράμετρο *id* (δηλ., $id = \text{nonce}$), ενώ το μέγεθος της ετικέτας είναι 128 bits (ή $t = 16$ bytes). Κατά την διαδικασία της κρυπτογράφησης ενός απλού κειμένου P , με ένα δεδομένο μοναδικό αριθμό *nonce*, συσχετισμένα δεδομένα A , με χρήση ενός κλειδιού K , και λαμβάνοντας ως έξοδο μια ετικέτα μεγέθους $t = 16$ bytes, η σειρά των κλήσεων που πραγματοποιούνται είναι η ακόλουθη:

```
CYCLIST( $K, \text{nonce}, \epsilon$ )
ABSORB( $A$ )
 $C \leftarrow \text{ENCRYPT}(P)$ 
 $T \leftarrow \text{SQUEEZE}(t)$ 
return ( $C, T$ )
```

Για την αποκρυπτογράφηση του (C, T) , η σειρά των κλήσεων που πραγματοποιούνται είναι η εξής παρόμοια:

```
CYCLIST( $K, \text{nonce}, \epsilon$ )
ABSORB( $A$ )
 $P \leftarrow \text{DECRYPT}(C)$ 
 $T' \leftarrow \text{SQUEEZE}(t)$ 
if  $T = T'$  then
    return  $P$ 
else
    return  $\perp$ 
```

13.3.5.4 Λειτουργία Συνάρτησης Σύνοψης *Xoodyak*

Το σχήμα *XOODYAK* μπορεί επίσης να χρησιμοποιηθεί ως συνάρτηση σύνοψης. Από προεπιλογή, η αντιμετάθεση που χρησιμοποιείται είναι η *Xoodoo* [28] με μέγεθος 48 bytes (ή $b = 384$ bits) και το μέγεθος εξόδου είναι $n = 32$ bytes (ή 256 bits), ή αλλιώς το προεπιλεγμένο μέγεθος εξόδου επιλέγεται ελεύθερα όπως σε μια συνάρτηση επεκτάσιμης εξόδου (XOF). Για την λήψη μιας σύνοψης με μήκος n -byte για κάποια είσοδο δεδομένων x , το σχήμα *XOODYAK* μπορεί να χρησιμοποιηθεί με την ακόλουθη σειρά κλήσεων:

```
CYCLIST( $\epsilon, \epsilon, \epsilon$ )
ABSORB( $x$ )
 $H \leftarrow \text{SQUEEZE}(n)$ 
return  $H$ 
```

Βιβλιογραφία

- [1] Meltem Sönmez Turan et al. “Status Report on the Second Round of the NIST Lightweight Cryptography Standardization Process”. In: *National Institute of Standards and Technology (NIST) NISTIR 8369* (2021), pp. 1–81. doi: 10.6028/NIST.IR.8369.

- [2] Charalampos Manifavas, George Hatzivasilis, Konstantinos Fysarakis, and Konstantinos Rantos. “Lightweight Cryptography for Embedded Systems – A Comparative Analysis”. In: *Data Privacy Management and Autonomous Spontaneous Security*. Ed. by Joaquin Garcia-Alfaro, Georgios Lioudakis, Nora Cuppens-Boulahia, Simon Foley, and William M. Fitzgerald. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 333–349. ISBN: 978-3-642-54568-9. DOI: 10.1007/978-3-642-54568-9_21.
- [3] Rhys Weatherley. *LWC Finalists*. <https://rweather.github.io/lwc-finalists/>. Accessed January 27, 2022. 2021.
- [4] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. “ASCON v1.2: Lightweight Authenticated Encryption and Hashing”. In: *Journal of Cryptology* 34.3 (2021), p. 33. DOI: 10.1007/s00145-021-09398-9.
- [5] Tim Beyne, Yu Long Chen, Christoph Dobraunig, and Bart Mennink. “Multi-User Security of the Elephant v2 Authenticated Encryption Mode”. In: *Selected Areas in Cryptography – SAC 2021*. Ed. by Andreas Hülsing and Riham AlTawy. Cham: Springer International Publishing, 2022, pp. 1–24.
- [6] Subhadeep Banik et al. *GIFT-COFB*. Cryptology ePrint Archive, Report 2020/738. <https://ia.cr/2020/738>. 2020.
- [7] Martin Hell, Thomas Johansson, Alexander Maximov, Willi Meier, and Hirotaka Yoshida. “Grain-128AEADv2: Strengthening the Initialization Against Key Reconstruction”. In: *Cryptology and Network Security*. Ed. by Mauro Conti, Marc Stevens, and Stephan Krenn. Cham: Springer International Publishing, 2021, pp. 24–41. ISBN: 978-3-030-92548-2. DOI: 10.1007/978-3-030-92548-2_2.
- [8] Christoph Dobraunig et al. “ISAP v2.0”. In: *IACR Transactions on Symmetric Cryptology* 2020.S1 (June 2020), pp. 390–416. DOI: 10.13154/tosc.v2020.iS1.390-416.
- [9] Zhenzhen Bao et al. *PHOTON-Beetle Authenticated Encryption and Hash Family*. Submission to the NIST Lightweight Cryptography Standardization Process. <https://www.isical.ac.in/~lightweight/beetle/>. 2021.
- [10] Chun Guo, Tetsu Iwata, Mustafa Khairallah, Kazuhiko Minematsu, and Thomas Peyrin. *Romulus v1.3*. Submission to the NIST Lightweight Cryptography Standardization Process. <https://romulusae.github.io/romulus/>. 2021.
- [11] Christof Beierle et al. *SCHWAEMM and ESCH: Lightweight authenticated encryption and hashing using the SPARKLE permutation family*. Submission to the NIST Lightweight Cryptography Standardization Process. <https://sparkle-lwc.github.io/>. 2021.
- [12] Hongjun Wu and Tao Huang. *TinyJAMBU: A Family of Lightweight Authenticated Encryption Algorithms (Version 2)*. Submission to the NIST Lightweight Cryptography Standardization Process. 2021.
- [13] Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. “Xoodyak, a lightweight cryptographic scheme”. In: *IACR Transactions on Symmetric Cryptology* 2020.S1 (June 2020), pp. 60–87. DOI: 10.13154/tosc.v2020.iS1.60-87.
- [14] Subhadeep Banik et al. “GIFT: A Small Present”. In: *Cryptographic Hardware and Embedded Systems – CHES 2017*. Ed. by Wieland Fischer and Naofumi Homma. Cham: Springer International Publishing, 2017, pp. 321–345. ISBN: 978-3-319-66787-4. DOI: 10.1007/978-3-319-66787-4_16.

- [15] Avik Chakraborti, Tetsu Iwata, Kazuhiko Minematsu, and Mridul Nandi. “Blockcipher-Based Authenticated Encryption: How Small Can We Go?” In: *Cryptographic Hardware and Embedded Systems – CHES 2017*. Ed. by Wieland Fischer and Naofumi Homma. Cham: Springer International Publishing, 2017, pp. 277–298. ISBN: 978-3-319-66787-4. doi: 10.1007/978-3-319-66787-4_14.
- [16] Andrey Bogdanov et al. “PRESENT: An Ultra-Lightweight Block Cipher”. In: *Cryptographic Hardware and Embedded Systems - CHES 2007*. Ed. by Pascal Paillier and Ingrid Verbauwhede. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 450–466. ISBN: 978-3-540-74735-2. doi: 10.1007/978-3-540-74735-2_31.
- [17] Ray Beaulieu et al. “The SIMON and SPECK Lightweight Block Ciphers”. In: *Proceedings of the 52nd Annual Design Automation Conference*. DAC ’15. San Francisco, California: Association for Computing Machinery, 2015. ISBN: 9781450335201. doi: 10.1145/2744769.2747946.
- [18] Christof Beierle et al. “The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS”. In: *Advances in Cryptology – CRYPTO 2016*. Ed. by Matthew Robshaw and Jonathan Katz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 123–153. ISBN: 978-3-662-53008-5. doi: 10.1007/978-3-662-53008-5_5.
- [19] Christof Beierle et al. “Lightweight AEAD and Hashing using the SPARKLE Permutation Family”. In: *IACR Transactions on Symmetric Cryptology* 2020.S1 (June 2020), pp. 208–261. doi: 10.13154/tosc.v2020.iS1.208-261.
- [20] Christof Beierle et al. “Alzette: A 64-Bit ARX-box”. In: *Advances in Cryptology – CRYPTO 2020*. Ed. by Daniele Micciancio and Thomas Ristenpart. Cham: Springer International Publishing, 2020, pp. 419–448. ISBN: 978-3-030-56877-1. doi: 10.1007/978-3-030-56877-1_15.
- [21] Avik Chakraborti, Nilanjan Datta, Mridul Nandi, and Kan Yasuda. “Beetle Family of Lightweight and Secure Authenticated Encryption Ciphers”. In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* 2018.2 (May 2018), pp. 218–241. doi: 10.13154/tches.v2018.i2.218-241.
- [22] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. “Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications”. In: *Selected Areas in Cryptography*. Ed. by Ali Miri and Serge Vaudenay. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 320–337. ISBN: 978-3-642-28496-0. doi: 10.1007/978-3-642-28496-0_19.
- [23] Hongjun Wu and Tao Huang. *The JAMBU Lightweight Authentication Encryption Mode (v2.1)*. Submission to the CAESAR competition. <https://competitions.cr.yp.to/caesar.html>. 2016.
- [24] Joan Daemen, Bart Mennink, and Gilles Van Assche. “Full-State Keyed Duplex with Built-In Multi-user Support”. In: *Advances in Cryptology – ASIACRYPT 2017*. Ed. by Tsuyoshi Takagi and Thomas Peyrin. Cham: Springer International Publishing, 2017, pp. 606–637. ISBN: 978-3-319-70697-9. doi: 10.1007/978-3-319-70697-9_21.
- [25] Markku-Juhani O. Saarinen. “Beyond Modes: Building a Secure Record Protocol from a Cryptographic Sponge Permutation”. In: *Topics in Cryptology – CT-RSA 2014*. Ed. by Josh Benaloh. Cham: Springer International Publishing, 2014, pp. 270–285. ISBN: 978-3-319-04852-9. doi: 10.1007/978-3-319-04852-9_14.
- [26] Mike Hamburg. *The STROBE protocol framework*. Cryptology ePrint Archive, Report 2017/003. <https://ia.cr/2017/003>. 2017.
- [27] Trevor Perrin. *Stateful Hash Objects: API and Constructions*. GitHub, Version 4aef05a. https://github.com/noiseprotocol/sho_spec. 2018.

- [28] Joan Daemen, Seth Hoffert, Gilles Van Assche, and Ronny Van Keer. “The design of Xoodoo and Xoofff”. In: *IACR Transactions on Symmetric Cryptology* 2018.4 (Dec. 2018), pp. 1–38. doi: 10.13154/tosc.v2018.i4.1-38.
- [29] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. *The KECCAK reference*. <https://keccak.team/files/Keccak-reference-3.0.pdf>. 2011.
- [30] Penny Pritzker and Willie May. “SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions”. In: *Federal Information Processing Standards Publication FIPS PUB 202* (2015), pp. 1–29. doi: 10.6028/NIST.FIPS.202.
- [31] Daniel J. Bernstein et al. “Gimli : A Cross-Platform Permutation”. In: *Cryptographic Hardware and Embedded Systems – CHES 2017*. Ed. by Wieland Fischer and Naofumi Homma. Cham: Springer International Publishing, 2017, pp. 299–320. ISBN: 978-3-319-66787-4. doi: 10.1007/978-3-319-66787-4_15.
- [32] Guido Bertoni, Joan Daemen, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. *Keyak v2*. Submission to the CAESAR competition. <https://competitions.cr.yp.to/caesar.html>. 2016.

ΚΕΦΑΛΑΙΟ 14

ΕΦΑΡΜΟΣΜΕΝΗ ΚΒΑΝΤΙΚΗ ΚΡΥΠΤΟΓΡΑΦΙΑ

Περίληψη

Η κβαντική κρυπτογραφία αξιοποιεί ορισμένες βασικές κβαντομηχανικές ιδιότητες ενός φυσικού συστήματος με σκοπό την εκτέλεση κρυπτογραφικών εργασιών. Η βασική της διαφορά με την κλασσική κρυπτογραφία είναι ότι οι συμμετέχοντες στα κβαντικά κρυπτογραφικά πρωτόκολλα έχουν την δυνατότητα να καταλάβουν πότε και αν κάποιος υποκλέπτει την επικοινωνία τους καθώς η διαδικασία της μέτρησης από τους εισβολείς διαταράσσει το κβαντικό σύστημα προκαλώντας αναγνωρίσιμες από τους νόμιμους συμμετέχοντες αλλοιώσεις στις πληροφορίες, με αποτέλεσμα να εξασφαλίζεται μια απολύτως ασφαλής επικοινωνία. Πιο αναλυτικά, στην Ενότητα 14.1 δίνεται ο ορισμός της κβαντικής κρυπτογραφίας μαζί με μια ιστορική αναδρομή για την πρωτοεμφάνισή της, ενώ στην Ενότητα 14.2 γίνεται μια σύντομη εισαγωγή σε έννοιες της κβαντικής θεωρίας πληροφοριών για την καλύτερη κατανόηση των διαφόρων εφαρμογών της κβαντικής κρυπτογραφίας. Στην Ενότητα 14.3 παρουσιάζεται η κβαντική διανομή κλειδών, μιας από τις πιο γνωστές και πρακτικές εφαρμογές της κβαντικής κρυπτογραφίας. Στις Ενότητες 14.4, 14.5 και 14.6 παρουσιάζονται τρία αρκετά γνωστά κβαντικά κρυπτογραφικά πρωτόκολλα, αυτά της κβαντικής κοινής χρήσης μυστικών, της κβαντικής ρίψης νομίσματος και της κβαντικής δέσμευσης, αντίστοιχα. Τέλος, στις Ενότητες 14.7 και 14.8 αναφέρονται δύο πιο εξωτικά κβαντικά πρωτόκολλα, αυτό του οριοθετημένου και θορυβώδους μοντέλου κβαντικής αποθήκευσης, καθώς και αυτό της κβαντικής κρυπτογραφίας βάσει θέσης, ενώ το κεφάλαιο ολοκληρώνεται με τις σχετικές βιβλιογραφικές αναφορές.

Προαπαιτούμενη γνώση: Κατανόηση των βασικών αρχών της διαχείρισης κρυπτογραφικών κλειδιών (Κεφάλαιο 6) καθώς και κάποιων μηχανισμών ενίσχυσης του απορρήτου (Κεφάλαιο 8).

14.1 Ορισμός και Ιστορική Αναδρομή της Κβαντικής Κρυπτογραφίας

Η Κβαντική Κρυπτογραφία (Quantum Cryptography) είναι η επιστήμη που αξιοποιεί κβαντομηχανικές ιδιότητες για την εκτέλεση κρυπτογραφικών εργασιών [1, 2]. Το πιο γνωστό παράδειγμα κβαντικής κρυπτογραφίας είναι η Κβαντική Διανομή Κλειδών (Quantum Key Distribution – QKD) που προσφέρει μια

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx-978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

πληροφοριθεωρητικά ασφαλή λύση στο πρόβλημα της ανταλλαγής κλειδιών. Το πλεονέκτημα της κβαντικής κρυπτογραφίας έγκειται στο γεγονός ότι επιτρέπει την ολοκλήρωση διαφόρων κρυπτογραφικών εργασιών που αποδεδειγμένα ή συμπερασματικά είναι αδύνατες χρησιμοποιώντας μόνο κλασική (δηλαδή μη-κβαντική) επικοινωνία. Για παράδειγμα, είναι αδύνατο να αντιγραφούν δεδομένα που κωδικοποιούνται σε μια κβαντική κατάσταση. Και αυτό γιατί εάν κάποιος επιχειρήσει να διαβάσει τα κωδικοποιημένα δεδομένα, η κβαντική κατάσταση θα αλλάξει λόγω κατάρρευσης της κυματικής συνάρτησης (σύμφωνα με το θεώρημα της μη-κλωνοποίησης [3]). Αυτό θα μπορούσε να χρησιμοποιηθεί για την ανίχνευση υποκλοπών στην κβαντική διανομή κλειδιών.

Η έννοια της κβαντικής κρυπτογραφίας αρχικά εμφανίστηκε από τον Stephen Wiesner στις αρχές της δεκαετίας του 1970, όταν εργαζόταν στο Πανεπιστήμιο Columbia της Νέας Υόρκης, εισάγοντας την έννοια της κβαντικά συζευγμένης κωδικοποίησης (conjugate coding). Αξιοσημείωτο είναι ότι η εργασία του με τίτλο “Conjugate Coding” είχε απορριφθεί από την IEEE Information Theory Society αλλά τελικά δημοσιεύθηκε το 1983 στο SIGACT News [4]. Σε αυτήν την δημοσίευση έδειξε πώς να αποθηκεύονται ή να μεταδίδονται δύο μηνύματα κωδικοποιώντας τα ως δύο «συζευγμένες παρατηρήσιμες μετρήσεις» (conjugate observables), όπως η γραμμική και η κυκλική πόλωση των φωτονίων [5], έτσι ώστε ένα από τα δύο, αλλά όχι και τα δύο, να παραλαμβάνονται και να αποκωδικοποιούνται. Στην συνέχεια, ο Charles H. Bennett, του Ερευνητικού Κέντρου Thomas J. Watson της IBM, καθώς και ο Gilles Brassard, όταν συναντήθηκαν το 1979 στο 20th Symposium IEEE on the Foundations of Computer Science, που πραγματοποιήθηκε στο Πουέρτο Ρίκο, κατάλαβαν πώς να ερμηνεύσουν τα ευρήματα του Wiesner, συνειδητοποιώντας ότι τα φωτόνια δεν προορίζονταν να αποθηκεύονται πληροφορίες, αλλά να τις μεταδίδουν [4]. Το 1984, βασιζόμενοι σε αυτή την εργασία, οι Bennett και Brassard πρότειναν μια μέθοδο για ασφαλή επικοινωνία, η οποία ονομάζεται BB84 [6]. Ανεξάρτητα, το 1991 ο Artur Ekert πρότεινε να χρησιμοποιηθούν οι ανισότητες του Bell για την επίτευξη ασφαλούς διανομής κλειδιού [7]. Το πρωτόκόλλο του Ekert για τη διανομή κλειδιού, όπως παρουσιάστηκε στη συνέχεια από τους Dominic Mayers και Andrew Yao [8], προσφέρει κβαντική διανομή κλειδιών ανεξαρτήτως συσκευής (device-independent QKD). Αυτό σημαίνει ότι η ασφάλεια ενός πρωτοκόλλου δεν βασίζεται στην εμπιστοσύνη ότι οι κβαντικές συσκευές που χρησιμοποιούνται είναι πραγματικές, και επομένως η ανάλυση ασφαλείας ενός τέτοιου πρωτοκόλλου πρέπει να λαμβάνει υπόψη σενάρια ατελών ή ακόμη και κακόβουλων συσκευών. Τα χρόνια που ακολούθησαν εμφανίστηκαν διάφορες παραλαγές των βασικών πρωτοκόλλων της κβαντικής διανομής κλειδιών, αλλά και άλλων πρωτοκόλλων, όπως της κβαντικής ρίψης νομίσματος, της κβαντικής δέσμευσης, του οριοθετημένου και θορυβώδους μοντέλου κβαντικής αποθήκευσης, αλλά και της κβαντικής κρυπτογραφίας βάσει θέσης, που παρουσιάζονται σε επόμενες ενότητες αυτού του κεφαλαίου.

14.2 Κβαντική Θεωρία Πληροφοριών

Σε αυτή την ενότητα γίνεται μια σύντομη εισαγωγή σε έννοιες της κβαντικής θεωρίας πληροφοριών [9] για την καλύτερη κατανόηση των επόμενων ενοτήτων που αφορούν τις διάφορες εφαρμογές της κβαντικής κρυπτογραφίας.

14.2.1 Κβαντικά Bits

Από τον Shannon και την αρχή της θεωρίας πληροφοριών, το bit ήταν ο βασικός όρος στην κλασική πληροφορία. Οι καταστάσεις ενός bit είναι είτε 0 είτε 1. Σύμφωνα με την κλασική έννοια, στην κβαντική πληροφορία υπάρχει το qubit (συντομογραφία του quantum bit). Όπως και για το κλασικό bit, είναι δυνατές δύο καταστάσεις, η $|0\rangle$ και η $|1\rangle$. Αυτός ο ειδικός συμβολισμός $| \rangle$ ονομάζεται Dirac και αποτελεί τον τυπικό συμβολισμό για καταστάσεις στην κβαντομηχανική. Η κύρια διαφορά σε σχέση με το κλασικό bit, το οποίο δέχεται μόνο 0 ή 1, είναι ότι ένα qubit επιτρέπει επίσης καταστάσεις μεταξύ $|0\rangle$ και $|1\rangle$, οι οποίες ονομάζονται υπερθέσεις

(superpositions). Οι οποίες ορίζονται ως εξής:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (14.1)$$

όπου $\alpha, \beta \in \mathbb{C}$. Επειδή αυτοί οι παράγοντες είναι μιγαδικοί αριθμοί, η κατάσταση ενός qubit μπορεί να περιγραφεί ως ένα διάνυσμα σε έναν δισδιάστατο μιγαδικό διανυσματικό χώρο \mathbb{C}^2 , που ονομάζεται χώρος Hilbert. Οι καταστάσεις $|0\rangle$ και $|1\rangle$ αποτελούν την υπολογιστική βάση και είναι ορθοκανονικές μεταξύ τους (δηλ., είναι ορθογώνιες (το εσωτερικό τους γινόμενο είναι 0) και αποτελούν μοναδιαία διανύσματα), π.χ. $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ και $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

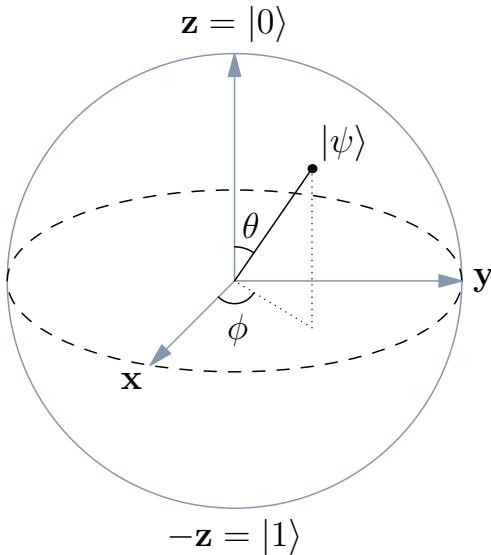
Δεδομένου ότι μια κατάσταση qubit είναι ένα μοναδιαίο διάνυσμα, πράγμα που σημαίνει ότι το μήκος του κανονικοποιείται στο 1, πρέπει να ισχύει η ακόλουθη εξίσωση για τις βαθμίδες (scalars) α, β :

$$|\alpha|^2 + |\beta|^2 = 1 \quad (14.2)$$

Με δεδομένη την εξίσωση 14.2, μπορούμε να ξαναγράψουμε την κατάσταση ενός qubit ως εξής:

$$|\psi\rangle = \cos \frac{\theta}{2}|0\rangle + e^{i\phi} \sin \frac{\theta}{2}|1\rangle \quad (14.3)$$

όπου i υποδηλώνει τη φανταστική μονάδα του μιγαδικού αριθμού, θ, ϕ είναι πραγματικοί αριθμοί και ορίζονται ένα σημείο σε μια σφαίρα που ονομάζεται σφαίρα Bloch (Σχήμα 14.1).



Σχήμα 14.1: Αναπαράσταση ενός qubit στην σφαίρα Bloch.

Η μέτρηση των qubits ωστόσο δεν είναι εύκολη υπόθεση. Στην ειδική περίπτωση που το α ή το β είναι 0, η αντιστοίχιση ενός qubit στο κλασικό bit θα έχει ως αποτέλεσμα 1 ή 0, αντίστοιχα, κάτι το οποίο είναι προφανές. Ωστόσο, στην περίπτωση που το qubit βρίσκεται σε άλλη υπέρθεση, δηλ. $\alpha, \beta \neq 0$, τότε το qubit θα μετρηθεί ως 1 με μια συγκεκριμένη πιθανότητα ή ως 0 με τη συμπληρωματική πιθανότητα. Εφόσον οι βαθμίδες πληρούν την Εξίσωση 14.2, η πιθανότητα για ένα qubit να μετρηθεί ως 0 είναι $|\alpha|^2$ και ως 1 είναι $|\beta|^2$.

Επιπλέον στην κβαντομηχανική οι βαθμίδες α και β ονομάζονται επίσης πλάτος (amplitude) των καταστάσεων $|0\rangle$ και $|1\rangle$, αντίστοιχα. Άλλα επιπλέον υπάρχει και ένας δεύτερος όρος που περιγράφει ένα qubit, η φάση (phase). Θεωρήστε την κατάσταση $e^{i\phi}|\psi\rangle$, όπου $|\psi\rangle$ είναι ένα διάνυσμα κατάστασης, και ϕ είναι ένας

πραγματικός αριθμός. Η κατάσταση $e^{i\phi}|\psi\rangle$ αναφέρεται ότι είναι ίση με $|\psi\rangle$, μέχρι τον συντελεστή καθολικής φάσης (global phase) $e^{i\phi}$.

Ένα άλλο είδος φάσης είναι η σχετική φάση (relative phase), η οποία αποτυπώνεται στις ακόλουθες δύο καταστάσεις:

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{και} \quad |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad (14.4)$$

Στην κατάσταση $|+\rangle$ το πλάτος του $|1\rangle$ είναι $\frac{1}{\sqrt{2}}$. Στην κατάσταση $|-\rangle$ το πλάτος έχει το ίδιο μέγεθος αλλά διαφορετικό πρόσημο. Επίσης, ορίζουμε ότι δύο πλάτη α_1, α_2 για ορισμένες καταστάσεις διαφέρουν κατά μια σχετική φάση εάν υπάρχει ένα πραγματικό ϕ τέτοιο ώστε $\alpha_1 = e^{i\phi}\alpha_2$. Σε αντίθεση με την καθολική φάση, όπου και τα δύο πλάτη της κατάστασης είναι διαφορετικά με βάση τον συντελεστή $e^{i\phi}$, στην σχετική φάση διαφέρει μόνο το ένα πλάτος με βάση τον συντελεστή $e^{i\phi}$.

14.2.2 Γραμμικοί Τελεστές

Η αλλαγή κατάστασης των qubits πραγματοποιείται από γραμμικούς τελεστές (linear operators). Ως εκ τούτου χρησιμοποιείται μια συνάρτηση A , λαμβάνοντας διανύσματα από το V στο W (V και W είναι διανυσματικοί χώροι του \mathbb{C}^*). Ο πιο βολικός τρόπος για να περιγράψουμε μια τέτοια συνάρτηση είναι η αναπαράστασή της ως πίνακας. Αν ο πίνακας A έχει m στήλες και n γραμμές και αυτός ο πίνακας πολλαπλασιαστεί με το διάνυσμα $|v\rangle \in \mathbb{C}^n$, προκύπτει ένα νέο διάνυσμα $|w\rangle \in \mathbb{C}^m$. Ο ισχυρισμός για έναν τέτοιον πίνακα A είναι να πληρεί την ακόλουθη εξίσωση γραμμικότητας [10]:

$$A\left(\sum_i a_i |v_i\rangle\right) = \sum_i a_i A |v_i\rangle \quad (14.5)$$

Έστω ότι $A : V \rightarrow W$ αποτελεί έναν γραμμικό τελεστή και $|v_1\rangle, \dots, |v_n\rangle$ αποτελούν βάση του V και $|w_1\rangle, \dots, |w_m\rangle$ βάση του W . Τότε υπάρχουν μιγαδικοί αριθμοί A_{1j}, \dots, A_{mj} , τέτοιοι ώστε:

$$A|v_j\rangle = \sum_i A_{ij} |w_i\rangle \quad \text{με } 1 \leq i \leq m, 1 \leq j \leq n \quad (14.6)$$

που συνθέτουν την αναπαράσταση πίνακα του τελεστή A .

Αντίθετα, ένας πίνακας $n \times m$ μπορεί να γίνει κατανοητός ως ο αντίθετος γραμμικός τελεστής που μετασχηματίζει διανύσματα από τον διανυσματικό χώρο W στον διανυσματικό χώρο V εκτελώντας τον πολλαπλασιασμό του πίνακα με αυτά τα διανύσματα.

Τέσσερις τέτοιοι γραμμικοί τελεστές είναι ενδεικτικά οι εξής:

- Οι πίνακες Pauli:** Αυτοί είναι πίνακες 2 επί 2 και αντιπροσωπεύουν ορισμένες απαραίτητες αλλαγές κατάστασης των qubits. Οι πίνακες αυτοί είναι οι εξής:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad \text{και} \quad I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (14.7)$$

Οι τελεστές Pauli X και Z είναι επίσης γνωστοί ως τελεστές αναστροφής bit (π.χ. $X|0\rangle$ και $X|1\rangle$) και αναστροφής φάσης (π.χ. $Z|+\rangle$ και $Z|-\rangle$), αντίστοιχα. Μια από τις δυνατότητες του Y είναι ο πολλαπλασιασμός του με τη φανταστική μονάδα i , δηλ. iY , που μπορεί να εκτελέσει ταυτόχρονα και τις δύο αναστροφές, μια αναστροφή bit και μια αναστροφή φάσης, αφού $iY = ZX$. Ο τελεστής Pauli I είναι ο ταυτοτικός πίνακας I , ο οποίος αφήνει αμετάβλητο τον πίνακα που πολλαπλασιάζεται.

- Το εσωτερικό γινόμενο:** Ένα εσωτερικό γινόμενο $\langle v|w\rangle$ είναι μια συνάρτηση που παίρνει ως είσοδο δύο διανύσματα $|v\rangle$ και $|w\rangle$ από το διανυσματικό χώρο V και παράγει έναν μιγαδικό αριθμό ως έξοδο. Για

παράδειγμα, το εσωτερικό γινόμενο δύο n -διάστατων διανυσμάτων στο πεδίο των μιγαδικών αριθμών ορίζεται ως εξής:

$$\langle v|w \rangle = \sum_i a_i^* b_i = (a_1^* \cdots a_n^*) \cdot \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} \quad (14.8)$$

- **Το εξωτερικό γινόμενο:** Το εξωτερικό γινόμενο δύο διανυσμάτων είναι ο αντίθετος πολλαπλασιασμός του εσωτερικού γινόμενου. Σε αντίθεση με το εσωτερικό γινόμενο που έχει ως αποτέλεσμα μια ενιαία μιγαδική τιμή, το εξωτερικό γινόμενο έχει ως αποτέλεσμα έναν πίνακα:

$$|v\rangle\langle w| = A_{i,j} = \begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix} \cdot (b_1^* \cdots b_n^*) = \begin{pmatrix} a_1 b_1^* & \cdots & a_1 b_n^* \\ \vdots & \ddots & \vdots \\ a_m b_1^* & \cdots & a_m b_n^* \end{pmatrix} \quad (14.9)$$

- **Το τανυστικό γινόμενο:** Το τανυστικό γινόμενο είναι μια λειτουργία για τη δημιουργία ενός μεγαλύτερου διανυσματικού χώρου από δύο μικρότερους διανυσματικούς χώρους. Με δεδομένο ότι έχουμε δύο διανυσματικούς χώρους V και W με διαστάσεις m και n , αντίστοιχα, τότε το $V \otimes W$ είναι ένας διανυσματικός χώρος mn , τα στοιχεία του οποίου είναι γραμμικοί συνδυασμοί τανυστικών γινομένων των στοιχείων $|v\rangle \in V$ και $|w\rangle \in W$.

14.2.3 Κβαντική Μέτρηση

Αυτή η ενότητα παρέχει τον τρόπο με τον οποίο μπορεί να περιγραφεί η επίδραση των μετρήσεων στα κβαντικά συστήματα με αναφορά στα τρία σημαντικά αξιώματα [10]. Αρχικά, για να μπορέσουμε να μετρήσουμε ένα κβάντο, πρέπει να ορίσουμε την περιοχή στην οποία λαμβάνει χώρα η κβαντομηχανική.

Αξίωμα 1: Ένας σύνθετος διανυσματικός χώρος με εσωτερικό γινόμενο (ονομάζεται επίσης χώρος Hilbert) σχετίζεται με οποιοδήποτε απομονωμένο φυσικό σύστημα. Αυτό είναι επίσης γνωστό ως ο χώρος καταστάσης του συστήματος. Με τα μοναδιαία διανύσματα του χώρου καταστάσεων του συστήματος (διανύσματα καταστάσεων) μπορούμε να καλύψουμε ολόκληρο το σύστημα.

Για να λάβουμε περισσότερες πληροφορίες για ένα συγκεκριμένο σύστημα θα μετρούσαμε τον χώρο καταστάσεων του συστήματος. Κάτι τέτοιο όμως δεν είναι εφικτό στην κβαντομηχανική, γιατί δεν μπορούμε να μετρήσουμε ποιος είναι ο χώρος καταστάσεων του συστήματος, ούτε μπορούμε να πούμε ποιο είναι το διάνυσμα καταστάσεων αυτού του συστήματος. Το απλούστερο και πιο σημαντικό σύστημα είναι το qubit (Ενότητα 14.2.1). Το επόμενο αξίωμα δίνει την περιγραφή του πώς οι καταστάσεις αλλάζουν με το χρόνο.

Αξίωμα 2: Η εξέλιξη ενός κλειστού κβαντικού συστήματος περιγράφεται από έναν ενιαίο μετασχηματισμό (unitary transformation). Δηλαδή, η κατάσταση $|\psi\rangle$ του συστήματος τη χρονική στιγμή t_1 σχετίζεται με την κατάσταση $|\psi'\rangle$ του συστήματος τη στιγμή t_2 σύμφωνα με έναν ενιαίο τελεστή U που εξαρτάται μόνο από τους χρόνους t_1 και t_2 :

$$|\psi'\rangle = U|\psi\rangle \quad (14.10)$$

Εάν και η κβαντομηχανική δεν μας αφήνει να μετρήσουμε τον χώρο καταστάσεων του συστήματος και την κβαντική κατάστασή του, μας διασφαλίζει ωστόσο ποιοι ενιαίοι τελεστές U περιγράφουν την αλλαγή σε οποιοδήποτε κλειστό κβαντικό σύστημα. Τέτοιους είδους τελεστές είδαμε στην Ενότητα 14.2.2. Στήνοντας λοιπόν τις απαραίτητες βάσεις, μπορούμε να συνεχίσουμε με τη μέτρηση.

Αξίωμα 3: Οι κβαντικές μετρήσεις περιγράφονται από μια συλλογή M_m τελεστών μέτρησης. Αυτοί είναι τελεστές που ενεργούν στον χώρο καταστάσεων του συστήματος που μετράται. Ο δείκτης m αναφέρεται στα αποτελέσματα της μέτρησης (δηλ., 0 και 1) που μπορούν να προκύψουν στο πείραμα. Αν η κατάσταση του κβαντικού συστήματος είναι $|\psi\rangle$ αμέσως πριν τη μέτρηση, τότε η πιθανότητα να προκύψει το αποτέλεσμα m δίνεται από την εξίσωση:

$$p(m) = \langle \psi | M_m^\dagger M_m | \psi \rangle \quad (14.11)$$

όπου M_m^\dagger είναι το ερμιτιανό συζυγές (hermitian conjugate) του M_m , ενώ η κατάσταση του συστήματος μετά τη μέτρηση θα δίνεται από την ακόλουθη εξίσωση:

$$|\psi'\rangle = \frac{M_m|\psi\rangle}{\sqrt{\langle\psi|M_m^\dagger M_m|\psi\rangle}} \quad (14.12)$$

Οι τελεστές μέτρησης ικανοποιούν την εξίσωση πληρότητας:

$$\sum_m M_m^\dagger M_m = I \quad (14.13)$$

και η εξίσωση πληρότητας εκφράζει το γεγονός ότι οι πιθανότητες αθροίζονται σε 1:

$$\sum_m p(m) = \sum_m \langle\psi|M_m^\dagger M_m|\psi\rangle = 1 \quad (14.14)$$

14.2.4 Το Θεώρημα της Μη-Κλωνοποίησης

Το θεώρημα της μη-κλωνοποίησης παρουσιάστηκε για πρώτη φορά στο [3], σύμφωνα με το οποίο δεν είναι δυνατόν να δημιουργήσουμε ένα αντίγραφο μιας άγνωστης κβαντικής κατάστασης.

Ας υποθέσουμε ότι έχουμε μια κβαντική μηχανή με δύο υποδοχές με την ένδειξη A και B . Η υποδοχή A είναι η υποδοχή δεδομένων και ξεκινά σε μια κβαντική κατάσταση $|\psi\rangle$, χωρίς να ξέρουμε ποια ακριβώς κατάσταση έχει. Ο στόχος είναι να αντιγράψουμε την κατάσταση στην υποδοχή B , δηλαδή την υποδοχή στόχου. Υποθέτουμε ότι η υποδοχή στόχου ξεκινά σε κάποια ανεξάρτητη κατάσταση $|s\rangle$. Οπότε η αρχική κατάσταση του αντιγραφικού συστήματος ορίζεται ως εξής:

$$|\psi\rangle \otimes |s\rangle \quad (14.15)$$

Κάποια ενιαία εξέλιξη (unitary evolution) U επηρεάζει τώρα τη διαδικασία αντιγραφής, και ιδανικά γίνεται το εξής:

$$|\psi\rangle \otimes |s\rangle \rightarrow_U U(|\psi\rangle \otimes |s\rangle) = |\psi\rangle \otimes |\psi\rangle \quad (14.16)$$

Αν υποθέσουμε ότι αυτή η διαδικασία αντιγραφής λειτουργεί για δύο συγκεκριμένες καταστάσεις, $|\psi\rangle$ και $|\phi\rangle$, τότε έχουμε:

$$U(|\psi\rangle \otimes |s\rangle) = |\psi\rangle \otimes |\psi\rangle \quad \text{και} \quad U(|\phi\rangle \otimes |s\rangle) = |\phi\rangle \otimes |\phi\rangle \quad (14.17)$$

Λαμβάνοντας το εσωτερικό γινόμενο αυτών των δύο εξισώσεων προκύπτει το εξής:

$$\langle\psi|\phi\rangle = (\langle\psi|\phi\rangle)^2 \quad (14.18)$$

Αλλά το $x = x^2$ έχει μόνο δύο λύσεις, $x = 0$ και $x = 1$, οπότε είτε $|\psi\rangle = |\phi\rangle$, είτε $|\psi\rangle$ και $|\phi\rangle$ είναι ορθογώνια μεταξύ τους (δηλ., το εσωτερικό τους γινόμενο είναι 0). Εάν υπήρχε μια συσκευή κλωνοποίησης, θα μπορούσε να κλωνοποιήσει μόνο καταστάσεις που είναι ορθογώνιες μεταξύ τους, και επομένως μια γενική συσκευή κβαντικής κλωνοποίησης δεν μπορεί να υπάρξει. Για παράδειγμα, έχουμε qubits με καταστάσεις $|\psi\rangle$ και $|0\rangle$, όπου το ψ δεν είναι $|1\rangle$, επομένως είναι αδύνατο για έναν κβαντικό κλωνοποιητή να τα αντιγράψει, αφού αυτές οι καταστάσεις δεν είναι ορθογώνιες.

14.3 Κβαντική Διανομή Κλειδιών

Η Κβαντική Διανομή Κλειδιών (Quantum Key Distribution – QKD) αποτελεί μια ασφαλή μέθοδο επικοινωνίας που εφαρμόζει ένα κρυπτογραφικό πρωτόκολλο που περιλαμβάνει στοιχεία της κβαντικής μηχανικής. Επιτρέπει σε δύο μέρη να παράγουν ένα κοινό τυχαίο μυστικό κλειδί που είναι γνωστό μόνο σε αυτά, και

το οποίο στη συνέχεια μπορεί να χρησιμοποιηθεί για την κρυπτογράφηση και την αποκρυπτογράφηση μηνυμάτων. Συχνά, λανθασμένα ωστόσο, ονομάζεται κβαντική κρυπτογραφία, καθώς είναι το πιο γνωστό και εφαρμοσμένο παράδειγμα κβαντικής κρυπτογραφικής εργασίας.

Μια σημαντική και μοναδική ιδιότητα της κβαντικής διανομής κλειδιών είναι η ικανότητα των δύο συμμετέχοντων που επικοινωνούν να ανιχνεύουν την παρουσία οποιουδήποτε τρίτου που προσπαθεί να αποκτήσει γνώση του κλειδιού. Αυτό προκύπτει από μια θεμελιώδη ιδιότητα της κβαντικής μηχανικής: η διαδικασία μέτρησης ενός κβαντικού συστήματος γενικά διαταράσσει το σύστημα (βλέπε Ενότητα 14.2.3). Ένα τρίτο μέρος που προσπαθεί να κρυφακούσει το κλειδί πρέπει με κάποιο τρόπο να το μετρήσει, εισάγοντας έτσι ανιχνεύσιμες ανωμαλίες. Χρησιμοποιώντας κβαντικές υπερθέσεις ή κβαντική εμπλοκή και μεταδίδοντας πληροφορίες σε κβαντικές καταστάσεις, μπορεί να εφαρμοστεί ένα σύστημα επικοινωνίας που να ανιχνεύει την υποκλοπή πληροφοριών. Εάν το επίπεδο υποκλοπής είναι κάτω από ένα συγκεκριμένο όριο, μπορεί να δημιουργηθεί ένα κλειδί που είναι εγγυημένο ότι είναι ασφαλές (δηλαδή, ο υποκλοπέας δεν έχει πληροφορίες σχετικά με αυτό), διαφορετικά δεν είναι δυνατόν να παραχθεί ασφαλές κλειδί και η επικοινωνία ματαιώνεται.

Η ασφάλεια της κρυπτογράφησης που χρησιμοποιεί τη κβαντική διανομή κλειδιών βασίζεται στις θεμελιώσεις της κβαντικής μηχανικής, σε αντίθεση με την παραδοσιακή κρυπτογραφία δημόσιου κλειδιού, η οποία βασίζεται στην υπολογιστική δυσκολία ορισμένων μαθηματικών συναρτήσεων, και δεν μπορεί να παρέχει καμία μαθηματική απόδειξη ως προς την πραγματική πολυπλοκότητα της αντιστροφής των μονόδρομων συναρτήσεων που χρησιμοποιούνται. Η κβαντική διανομή κλειδιών παρέχει αποδειγμένη ασφάλεια με βάση τη θεωρία πληροφοριών και την προς τα εμπρός μυστικότητα (forward secrecy).

Το κύριο μειονέκτημα της κβαντικής διανομής κλειδιών είναι ότι συνήθως βασίζεται στην ύπαρξη ενός πιστοποιημένου κλασικού καναλιού επικοινωνίας. Στη σύγχρονη κρυπτογραφία, η ύπαρξη ενός πιστοποιημένου κλασικού καναλιού σημαίνει ότι κάποιος είτε έχει ήδη ανταλλάξει ένα συμμετρικό κλειδί κρυπτογράφησης επαρκούς μήκους, είτε δημόσια κλειδιά κρυπτογράφησης επαρκούς επιπέδου ασφαλείας. Με τέτοιες πληροφορίες ήδη διαθέσιμες, στην πράξη μπορεί κανείς να επιτύχει επαληθευμένες και επαρκώς ασφαλείς επικοινωνίες χωρίς τη χρήση της κβαντικής διανομής κλειδιών, όπως με τη χρήση της λειτουργίας counter mode (CTR) του κρυπτοσυστήματος AES.

Η κβαντική διανομή κλειδιών χρησιμοποιείται μόνο για την παραγωγή και τη διανομή ενός κλειδιού και όχι για τη μετάδοση δεδομένων μηνύματος. Αυτό το κλειδί μπορεί στη συνέχεια να χρησιμοποιηθεί με οποιονδήποτε συμβατικό αλγόριθμο κρυπτογράφησης για την κρυπτογράφηση (και αποκρυπτογράφηση) ενός μηνύματος, το οποίο στη συνέχεια μπορεί να μεταδοθεί μέσω ενός τυπικού καναλιού επικοινωνίας. Ο αλγόριθμος που συσχετίζεται πιο συχνά με την κβαντική διανομή κλειδιών είναι η κρυπτογράφηση one-time pad, καθώς είναι αποδειγμένα ασφαλής όταν χρησιμοποιείται με ένα μυστικό και τυχαίο κλειδί [11]. Σε πραγματικές καταστάσεις, χρησιμοποιείται συχνά και με συμμετρικούς αλγορίθμους κρυπτογράφησης, όπως ο αλγόριθμος AES (βλέπε Ενότητα 1.3.2).

Στις υποενότητες που ακολουθούν παρουσιάζονται δύο από τα πιο σημαντικά και γνωστά πρωτόκολλα κβαντικής διανομής κλειδιών, το BB84 [6] και το E91 [7].

14.3.1 Το Πρωτόκολλο BB84

Το πρωτόκολλο BB84, που πήρε το όνομα του από τα αρχικά των ονομάτων του Bennett και Brassard και έτος δημοσίευσής του το 1984, αρχικά περιγράφηκε χρησιμοποιώντας καταστάσεις πόλωσης φωτονίων για τη μετάδοση των πληροφοριών [6]. Ωστόσο, οποιαδήποτε δύο ζεύγη συζευγμένων καταστάσεων μπορούν να χρησιμοποιηθούν για το πρωτόκολλο, και πολλές υλοποιήσεις του που βασίζονται σε οπτικές ίνες χρησιμοποιούν καταστάσεις κωδικοποίησης φάσης. Ο αποστολέας και ο παραλήπτης συνδέονται μέσω ενός κβαντικού καναλιού επικοινωνίας που επιτρέπει τη μετάδοση κβαντικών καταστάσεων. Στην περίπτωση των φωτονίων αυτό το κανάλι είναι γενικά είτε μια οπτική ίνα, είτε απλά ο ελεύθερος χώρος. Επιπλέον, επικοινωνούν μέσω ενός δημόσιου κλασικού καναλιού, για παράδειγμα χρησιμοποιώντας ραδιοφωνικές συχνότητες ή το Διαδίκτυο. Το πρωτόκολλο έχει σχεδιαστεί με την υπόθεση ότι ένας υποκλοπέας μπορεί να παρέμβει με οποιονδήποτε τρόπο στο κβαντικό κανάλι, ενώ το κλασικό κανάλι πρέπει να αυθεντικοποιημένο [12].

Η ασφάλεια του πρωτοκόλλου προέρχεται από την κωδικοποίηση των πληροφοριών σε μη ορθογώνιες καταστάσεις. Η κβαντική απροσδιοριστία σημαίνει ότι αυτές οι καταστάσεις δεν μπορούν γενικά να μετρηθούν χωρίς να διαταραχθεί η αρχική κατάσταση (βλέπε το θεώρημα της μη-κλωνοποίησης στην Ενότητα 14.2.4). Το πρωτόκολλο BB84 χρησιμοποιεί δύο ζεύγη καταστάσεων, με κάθε ζεύγος να είναι συζευγμένο με το άλλο ζεύγος, και οι δύο καταστάσεις εντός του ενός ζεύγους να είναι ορθογώνιες μεταξύ τους. Ζεύγη ορθογωνικών καταστάσεων αναφέρονται ως βάση. Τα συνήθη ζεύγη καταστάσεων πόλωσης που χρησιμοποιούνται είναι είτε η ευθύγραμμη βάση της κατακόρυφης (0°) και της οριζόντιας (90°), είτε η διαγώνιος βάση των 45° και 135° μοιρών, ή η κυκλική βάση της αριστερόστροφης και της δεξιόστροφης πόλωσης. Οποιεσδήποτε δύο από αυτές τις βάσεις είναι συζευγμένες (conjugated) μεταξύ τους και έτσι οποιεσδήποτε δύο μπορούν να χρησιμοποιηθούν στο πρωτόκολλο. Παρακάτω χρησιμοποιούνται οι ευθύγραμμες και οι διαγώνιες βάσεις.

Το πρώτο βήμα στο πρωτόκολλο BB84 είναι η κβαντική μετάδοση, κατά την οποία γίνονται τα εξής:

1. Η Αλίκη δημιουργεί ένα τυχαίο bit (0 ή 1) και στη συνέχεια επιλέγει τυχαία μία από τις δύο βάσεις (ευθύγραμμη ή διαγώνια σε αυτήν την περίπτωση) για να το μεταδώσει.
2. Στη συνέχεια προετοιμάζει μια κατάσταση πόλωσης φωτονίων ανάλογα με την τιμή και τη βάση του bit, όπως φαίνεται στον Πίνακα 14.1. Έτσι, για παράδειγμα, το 0 κωδικοποιείται στην ευθύγραμμη βάση (+) ως κατάσταση κατακόρυφης πόλωσης, και το 1 κωδικοποιείται στη διαγώνια βάση (x) ως κατάσταση 135° μοιρών.
3. Στη συνέχεια, η Αλίκη μεταδίδει ένα μόνο φωτόνιο στην κατάσταση που έχει καθοριστεί στον Μπάμπη, χρησιμοποιώντας το κβαντικό κανάλι.

Αυτή η διαδικασία επαναλαμβάνεται από το βήμα 1 έως και 3, με την Alice να καταγράφει την κατάσταση, τη βάση και τον χρόνο που αποστέλλεται κάθε φωτόνιο.

Πίνακας 14.1: Κατάσταση πόλωσης φωτονίων ανάλογα με την τιμή και τη βάση ενός bit.

Βάση	Bits	
	0	1
Ευθύγραμμη βάση (+)	↑	→
Διαγώνιος βάση (x)	↗	↘

Σύμφωνα με την κβαντομηχανική (ιδιαίτερα την κβαντική απροσδιοριστία), καμία δυνατή μέτρηση δεν μπορεί να κάνει διάκριση μεταξύ των 4 διαφορετικών καταστάσεων πόλωσης, καθώς δεν είναι όλες ορθογώνιες. Η μόνη δυνατή μέτρηση είναι μεταξύ οποιωνδήποτε δύο ορθογώνιων καταστάσεων (μια ορθοκανονική βάση). Έτσι, για παράδειγμα, η μέτρηση στην ευθύγραμμη βάση δίνει ένα αποτέλεσμα οριζόντιας ή κατακόρυφης. Εάν το φωτόνιο δημιουργήθηκε ως οριζόντιο ή κατακόρυφο (ως ευθύγραμμη ιδιοκατάσταση (eigenstate)) τότε αυτό μετρά τη σωστή κατάσταση, αλλά εάν δημιουργήθηκε ως 45° ή 135° μοιρών (διαγώνιες ιδιοκαταστάσεις), τότε η ευθύγραμμη μέτρηση επιστρέφει τυχαία είτε οριζόντια είτε κατακόρυφη. Επιπλέον, μετά από αυτή τη μέτρηση το φωτόνιο πολώνεται στην κατάσταση στην οποία μετρήθηκε (οριζόντια ή κατακόρυφη), με όλες τις πληροφορίες σχετικά με την αρχική του πόλωση να χάνονται.

Τα βήματα του πρωτοκόλλου που ακολουθούνται στην συνέχεια είναι τα εξής:

4. Καθώς ο Μπάμπης δεν γνωρίζει τη βάση στην οποία κωδικοποιήθηκαν τα φωτόνια, το μόνο που μπορεί να κάνει είναι να επιλέξει μια βάση τυχαία για μέτρηση, είτε ευθύγραμμη είτε διαγώνια. Αυτό το κάνει για κάθε φωτόνιο που λαμβάνει, καταγράφοντας τον χρόνο, τη βάση μέτρησης που χρησιμοποιήθηκε και το αποτέλεσμα της μέτρησης.
5. Αφού ο Μπάμπης μέτρησε όλα τα φωτόνια, επικοινωνεί με την Αλίκη μέσω του δημόσιου κλασικού καναλιού.

Πίνακας 14.2: Παράδειγμα εκτέλεσης του πρωτοκόλλου BB84.

Βήματα Εκτέλεσης	Χρονικές Στιγμές							
	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
Τυχαία bits της Αλίκης	0	1	1	0	1	0	0	1
Τυχαία επιλογή βάσης από την Αλίκη	+	+	\times	+	\times	\times	\times	+
Πόλωση φωτονίων που στέλνει η Αλίκη	\uparrow	\rightarrow	\searrow	\uparrow	\searrow	\nearrow	\nearrow	\rightarrow
Τυχαία βάση μετρησης του Μπάμπη	+	\times	\times	\times	+	\times	\times	+
Πόλωση φωτονίων που μετράει ο Μπάμπης	\uparrow	\nearrow	\searrow	\nearrow	\rightarrow	\nearrow	\rightarrow	\rightarrow
Δημόσια ενημέρωση για τις κοινές βάσεις μέσω κλασικού καναλιού								
Κοινό μυστικό κλειδί	0			1		0		1

6. Η Αλίκη αποστέλλει τη βάση στην οποία στάλθηκε κάθε φωτόνιο και ο Μπάμπης τη βάση στην οποία μετρήθηκε το καθένα.
7. Απορρίπτουν και οι δύο μετρήσεις φωτονίων (bits) όπου ο Μπάμπης χρησιμοποίησε διαφορετική βάση, οι οποίες είναι οι μισές κατά μέσο όρο, αφήνοντας τα μισά bits ως κοινό τους κλειδί.

Στον Πίνακα 14.2 παρατίθεται ένα παράδειγμα των βημάτων εκτέλεσης του πρωτοκόλλου BB84 για την αποστολή 8 αρχικών bits από την Αλίκη, από τα οποία τελικά προκύπτουν μόνο 4 κοινά μυστικά bits.

Για τον έλεγχο της παρουσίας ενός υποκλοπέα, η Αλίκη και ο Μπάμπης μπορούν να συγκρίνουν ένα προκαθορισμένο υποσύνολο των υπόλοιπων bits. Εάν ένας τρίτος (συνήθως αναφέρεται ως Εύα) έχει λάβει οποιαδήποτε πληροφορία σχετικά με την πόλωση των φωτονίων, αυτό εισάγει σφάλματα στις μετρήσεις του Μπάμπη. Επίσης, άλλες περιβαλλοντικές συνθήκες μπορούν να προκαλέσουν σφάλματα με παρόμοιο τρόπο. Εάν διαφέρουν περισσότερα από p bits, διακόπτουν την διανομή του κλειδιού και δοκιμάζουν ξανά, πιθανώς με διαφορετικό κβαντικό κανάλι, καθώς η ασφάλεια του κλειδιού δεν είναι εγγυημένη. Το p επιλέγεται με τέτοιο τρόπο ώστε εάν ο αριθμός των bits που είναι γνωστός στην Εύα είναι μικρότερος από αυτόν (δηλ., το p), η μέθοδος ενίσχυσης απορρήτου (privacy amplification) να μπορεί να χρησιμοποιηθεί για να μειωθεί η γνώση της Εύας για το κλειδί σε ένα αυθαίρετα μικρό ποσό με κόστος την μείωση του μήκους του κλειδιού.

14.3.2 Το Πρωτόκολλο E91

Το πρωτόκολλο E91, που πήρε και αυτό το όνομα του από το αρχικό του ονόματος του Ekert και το έτος δημοσίευσης του το 1991, χρησιμοποιεί μπερδεμένα (entangled) ζεύγη φωτονίων [7]. Αυτά μπορούν να δημιουργηθούν από την Αλίκη, τον Μπάμπη ή από κάποια διαφορετική πηγή και από αυτούς τους δύο, συμπεριλαμβανομένης της Εύας που δρα ως υποκλοπέας. Τα φωτόνια κατανέμονται έτσι ώστε η Αλίκη και ο Μπάμπης να καταλήγουν με ένα φωτόνιο από το κάθε ζεύγος.

Το πρωτόκολλο αυτό βασίζεται σε δύο ιδιότητες της κβαντικής εμπλοκής (entanglement). Πρώτον, οι μπερδεμένες καταστάσεις συσχετίζονται τέλεια με την έννοια ότι αν η Αλίκη και ο Μπάμπης μετρήσουν και οι δύο αν τα φωτόνια τους έχουν κατακόρυφες ή οριζόντιες πολώσεις, παίρνουν πάντα την ίδια απάντηση με 100% πιθανότητα. Το ίδιο ισχύει αν και οι δύο μετρήσουν οποιοδήποτε άλλο ζεύγος συμπληρωματικών (ορθογώνιων) πολώσεων. Αυτό προϋποθέτει ότι τα δύο απομακρυσμένα μέρη έχουν ακριβή συγχρονισμό κατευθυντικότητας. Ωστόσο, τα συγκεκριμένα αποτελέσματα είναι εντελώς τυχαία. Είναι αδύνατον για την Αλίκη να προβλέψει αν αυτή (και επομένως ο Μπάμπης) θα έχει κατακόρυφη ή οριζόντια πολώση. Δεύτερον, οποιαδήποτε προσπάθεια υποκλοπής από την Εύα επηρεάζει αυτούς τους συσχετισμούς με τρόπο που η Αλίκη και ο Μπάμπης μπορούν να ανιχνεύσουν.

Παρόμοια με το πρωτόκολλο BB84, το πρωτόκολλο E91 περιλαμβάνει ένα ιδιωτικό πρωτόκολλο μέτρησης

πριν από την ανίχνευση της παρουσίας της Εύας. Το στάδιο μέτρησης περιλαμβάνει την Αλίκη που μετρά κάθε φωτόνιο που λαμβάνει χρησιμοποιώντας κάποια βάση από το σύνολο $Z_0, Z_{\frac{\pi}{8}}, Z_{\frac{\pi}{4}}$, ενώ ο Μπάμπης επιλέγει βάση από το σύνολο $Z_0, Z_{\frac{\pi}{8}}, Z_{-\frac{\pi}{8}}$, όπου Z_θ είναι η βάση $\{|\uparrow\rangle, |\rightarrow\rangle\}$ με περιστροφή κατά θ . Και οι δύο διατηρούν μυστική τη σειρά των επιλογών βάσης μέχρι να ολοκληρωθούν οι μετρήσεις. Δημιουργούνται δύο ομάδες φωτονίων: η πρώτη αποτελείται από φωτόνια που μετρήθηκαν χρησιμοποιώντας την ίδια βάση από την Αλίκη και τον Μπάμπη, ενώ η δεύτερη περιέχει όλα τα άλλα φωτόνια. Για να ανιχνεύσουν τις υποκλοπές, μπορούν να υπολογίσουν τη στατιστική δοκιμή S χρησιμοποιώντας τους συντελεστές συσχέτισης μεταξύ των βάσεων της Αλίκης και των βάσεων του Μπάμπη, παρόμοιους με εκείνους που εμφανίζονται στα πειράματα δοκιμής Bell. Τα μέγιστα μπερδεμένα φωτόνια θα οδηγήσουν σε $|S| = 2\sqrt{2}$. Εάν δεν συμβεί αυτό, τότε η Αλίκη και ο Μπάμπης μπορούν να συμπεράνουν ότι η Εύα εισήγαγε τοπικό ρεαλισμό (local realism) στο σύστημα, παραβιάζοντας το θεώρημα του Bell [13]. Εάν το πρωτόκολλο ολοκληρωθεί με επιτυχία, η πρώτη ομάδα μπορεί να χρησιμοποιηθεί για τη δημιουργία κλειδιών, καθώς αυτά τα φωτόνια είναι εντελώς αντιευθυγραμμισμένα (anti-aligned) μεταξύ της Αλίκης και του Μπάμπη.

14.4 Κβαντική Κοινή Χρήση Μυστικών

Η κβαντική κοινή χρήση μυστικών (Quantum Secret Sharing – QSS) αποτελεί ένα κβαντικό κρυπτογραφικό σχήμα για ασφαλή επικοινωνία που επεκτείνεται πέρα από την απλή κβαντική διανομή κλειδιών. Αυτό τροποποιεί το κλασικό σχήμα κοινής χρήσης μυστικών χρησιμοποιώντας κβαντικές πληροφορίες και το θεώρημα της μη-κλωνοποίησης για την επίτευξη της απόλυτης ασφάλειας κατά την επικοινωνία.

Η μέθοδος της κοινής χρήσης μυστικών αποτελείται από έναν αποστολέα που επιθυμεί να μοιραστεί ένα μυστικό με έναν αριθμό παραληπτών με τέτοιο τρόπο ώστε το μυστικό να αποκαλύπτεται πλήρως μόνο εάν ένα αρκετά μεγάλο μέρος των παραληπτών συνεργαστεί. Ωστόσο, αν δεν συνεργαστούν αρκετοί παραλήπτες για να αποκαλύψουν το μυστικό, το μυστικό παραμένει εντελώς άγνωστο.

Το κλασικό σχήμα της κοινής χρήσης μυστικών προτάθηκε αρχικά και ανεξάρτητα από τον Adi Shamir [14] και τον George Blakley [15] το 1979. Το 1998, οι Hillery, Buzek και Berthiaume [16] επέκτειναν αυτήν τη θεωρία κάνοντας χρήση των κβαντικών καταστάσεων για τη δημιουργία ενός ασφαλούς κλειδιού που θα μπορούσε να χρησιμοποιείται για τη μετάδοση του μυστικού μέσω κλασικών δεδομένων (bits). Στα χρόνια που ακολούθησαν [17, 18], έγινε περισσότερη δουλειά σε αυτήν την περιοχή για να επεκταθεί η θεωρία στη μετάδοση κβαντικών πληροφοριών ως μυστικό, αντί να χρησιμοποιηθούν απλώς κβαντικές καταστάσεις για τη δημιουργία του κρυπτογραφικού κλειδιού.

Στις παραγράφους που ακολουθούν γίνεται παρουσίαση του αρχικού σχήματος που παρουσιάστηκε από τους Hillery *et al.* [16] και το οποίο κάνει χρήση των καταστάσεων Greenberger–Horne–Zeilinger (GHZ). Το πρωτόκολλο αυτό ουσιαστικά αποτελεί μια επέκταση της κβαντικής διανομής κλειδιών σε δύο παραλήπτες αντί μόνο σε έναν. Στο πρωτόκολλο αυτό ο αποστολέας υποδηλώνεται ως η Αλίκη και οι δύο παραλήπτες ως ο Μπάμπης και ο Τσάρλι. Ο στόχος της Αλίκης είναι να στείλει σε κάθε παραλήπτη ένα «μερίδιο» του μυστικού κλειδιού της (στην πραγματικότητα απλώς μια κβαντική κατάσταση) με τέτοιο τρόπο ώστε:

1. Ούτε το μερίδιο του Μπάμπη, ούτε του Τσάρλι να περιέχουν κάποια πληροφορία για το αρχικό μήνυμα της Αλίκης και επομένως κανένας από τους δύο να μην μπορεί να εξάγει το μυστικό μόνος του.
2. Το μυστικό μπορεί να εξαχθεί μόνον εάν ο Μπάμπης και ο Τσάρλι συνεργαστούν, οπότε το μυστικό αποκαλύπτεται πλήρως.
3. Η παρουσία είτε ενός εξωτερικού υποκλοπέα, είτε ενός ανέντιμου παραλήπτη (είτε του Μπάμπη είτε του Τσάρλι) μπορεί να εντοπιστεί χωρίς να αποκαλυφθεί το μυστικό.

Η Αλίκη ξεκινάει το πρωτόκολλο μοιράζοντας σε κάθε έναν από τους παραλήπτες, Μπάμπη και Τσάρλι, ένα

μερίδιο από μια τριπλέτα GHZ στην βάση Z, κρατώντας η ίδια το τρίτο μερίδιο:

$$|\Psi\rangle_{GHZ} = \frac{|000\rangle + |111\rangle}{\sqrt{2}} \quad (14.19)$$

όπου $|0\rangle$ και $|1\rangle$ είναι ορθογώνια σε έναν αυθαίρετο χώρο Hilbert.

Αφού ο κάθε συμμετέχων μετρήσει το μερίδιο του στη βάση X ή Y (που επιλέγεται τυχαία), μοιράζεται (μέσω ενός κλασικού δημόσιου καναλιού) ποια βάση χρησιμοποίησε για να κάνει τη μέτρηση, αλλά όχι το ίδιο το αποτέλεσμα. Συνδυάζοντας τα αποτελέσματα των μετρήσεών τους, ο Μπάμπης και ο Τσάρλι μπορούν να συμπεράνουν τι μέτρησε η Αλίκη στο 50% του χρόνου. Επαναλαμβάνοντας αυτή τη διαδικασία πολλές φορές και χρησιμοποιώντας ένα μικρό κλάσμα για να επαληθεύσουν ότι δεν υπάρχουν κακόβουλοι παράγοντες, οι τρεις συμμετέχοντες μπορούν να δημιουργήσουν ένα κοινό κλειδί για ασφαλή επικοινωνία. Ακολουθεί παρακάτω ένα παράδειγμα για το πώς θα λειτουργήσει αυτό.

Ας ορίσουμε τις ιδιοκαταστάσεις x και y με τον ακόλουθο τυπικό τρόπο:

$$\begin{aligned} |+x\rangle &= \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \quad |-x\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\ |+y\rangle &= \frac{|0\rangle + i|1\rangle}{\sqrt{2}}, \quad |-y\rangle = \frac{|0\rangle - i|1\rangle}{\sqrt{2}} \end{aligned}$$

Τότε, η κατάσταση GHZ μπορεί στη συνέχεια να ξαναγραφτεί ως εξής:

$$|\Psi\rangle_{GHZ} = \frac{1}{2\sqrt{2}} [(|+x\rangle_a|+x\rangle_b + |-x\rangle_a|-x\rangle_b)(|0\rangle_c + |1\rangle_c) + (|+x\rangle_a|-x\rangle_b + |+x\rangle_a|-x\rangle_b)(|0\rangle_c - |1\rangle_c)]$$

όπου τα a, b, c υποδηλώνουν τα μερίδια της Αλίκης, του Μπάμπη και του Τσάρλι, αντίστοιχα, και οι καταστάσεις της Αλίκης και του Μπάμπη έχουν γραφτεί στη βάση X. Χρησιμοποιώντας αυτή τη μορφή, είναι προφανές ότι υπάρχει μια συσχέτιση μεταξύ των μετρήσεων της Αλίκης και του Μπάμπη και της κατάστασης ενός μεριδίου του Τσάρλι: αν η Αλίκη και ο Μπάμπης έχουν συσχετισμένα αποτελέσματα, τότε ο Τσάρλι έχει την κατάσταση $\frac{|0\rangle_c + |1\rangle_c}{\sqrt{2}}$ και αν η Αλίκη και ο Μπάμπης έχουν αντι-συσχετισμένα αποτελέσματα τότε ο Τσάρλι έχει την κατάσταση $\frac{|0\rangle_c - |1\rangle_c}{\sqrt{2}}$.

Όπως φαίνεται και στον Πίνακα 14.3, ο οποίος συνοψίζει όλες αυτές τις συσχετίσεις, γνωρίζοντας τις βάσεις μέτρησης της Αλίκης και του Μπάμπη, ο Τσάρλι μπορεί να χρησιμοποιήσει το δικό του αποτέλεσμα μέτρησης για να συμπεράνει εάν η Αλίκη και ο Μπάμπης είχαν τα ίδια ή αντίθετα αποτελέσματα. Σημειώνεται, ωστόσο, ότι για να βγάλει αυτό το συμπέρασμα, ο Τσάρλι πρέπει να επιλέξει τη σωστή βάση μέτρησης για τη μέτρηση του δικού του μεριδίου. Εφόσον επιλέγει τυχαία ανάμεσα σε δύο βάσεις, μόνο τις μισές φορές θα μπορεί να εξάγει χρήσιμες πληροφορίες. Τις άλλες μισές φορές τα αποτελέσματα πρέπει να απορρίπτονται. Επιπλέον, από τον ίδιο πίνακα μπορεί κανείς να δει ότι ο Τσάρλι δεν έχει τρόπο να προσδιορίσει ποιος μέτρησε τι, παρά μόνον αν τα αποτελέσματα της Αλίκης και του Μπάμπη ήταν συσχετισμένα ή αντι-συσχετισμένα. Έτσι, ο μόνος τρόπος να καταλάβει ο Τσάρλι τη μέτρηση της Αλίκης είναι να συνεργαστεί με τον Μπάμπη και να μοιραστούν τα αποτελέσματά τους. Με αυτόν τον τρόπο, μπορούν να εξάγουν τα αποτελέσματα της Αλίκης για κάθε μέτρηση και να χρησιμοποιήσουν αυτές τις πληροφορίες για να δημιουργήσουν ένα κρυπτογραφικό κλειδί που μόνο αυτοί γνωρίζουν.

14.5 Κβαντική Ρίψη Νομίσματος

Φανταστείτε δύο απομακρυσμένους παίχτες, που συνδέονται μέσω ενός καναλιού επικοινωνίας, και δεν εμπιστεύονται ο ένας τον άλλον. Το πρόβλημα της συμφωνίας τους σε ένα τυχαίο bit ανταλλάσσοντας μηνύματα μέσω αυτού του καναλιού, χωρίς να βασίζονται σε κάποια έμπιστη τρίτη οντότητα, ονομάζεται το πρόβλημα

Πίνακας 14.3: Η επίδραση των μετρήσεων της Αλίκης και του Μπάμπη στην κατάσταση του Τσάρλι για την τριπλέτα GHZ.

		Αλίκη			
		+x	-x	+y	-y
Μπάμπης	+x	$ 0\rangle + 1\rangle$	$ 0\rangle - 1\rangle$	$ 0\rangle - i 1\rangle$	$ 0\rangle + i 1\rangle$
	-x	$ 0\rangle - 1\rangle$	$ 0\rangle + 1\rangle$	$ 0\rangle + i 1\rangle$	$ 0\rangle - i 1\rangle$
	+y	$ 0\rangle - i 1\rangle$	$ 0\rangle + i 1\rangle$	$ 0\rangle - 1\rangle$	$ 0\rangle + 1\rangle$
	-y	$ 0\rangle + i 1\rangle$	$ 0\rangle - i 1\rangle$	$ 0\rangle + 1\rangle$	$ 0\rangle - 1\rangle$

ρίψης νομίσματος στην κρυπτογραφία [19]. Η κβαντική ρίψη νομίσματος (quantum coin flipping) χρησιμοποιεί τις αρχές της κβαντικής μηχανικής στην κρυπτογράφηση μηνυμάτων για ασφαλή επικοινωνία. Αποτελεί ένα πρωτογενές κρυπτογραφικό εργαλείο που μπορεί να χρησιμοποιηθεί για την κατασκευή πιο περίπλοκων και χρήσιμων κρυπτογραφικών πρωτοκόλλων [20], όπως αυτό της κβαντικής βυζαντινής συμφωνίας [21].

Σε αντίθεση με άλλους τύπους κβαντικής κρυπτογραφίας (ιδιαίτερα της κβαντικής διανομής κλειδιού), η κβαντική ρίψη νομίσματος είναι ένα πρωτόκολλο που χρησιμοποιείται μεταξύ δύο χρηστών που δεν εμπιστεύονται ο ένας τον άλλον. Κατά συνέπεια, και οι δύο χρήστες (ή παίκτες) θέλουν να κερδίσουν την ρίψη του νομίσματος, ενώ θα προσπαθήσουν να εξαπατήσουν με διάφορους τρόπους [22].

Είναι γνωστό ότι εάν η επικοινωνία μεταξύ των παικτών γίνεται μέσω ενός κλασικού καναλιού, δηλαδή ενός καναλιού μέσω του οποίου δεν μπορούν να μεταδοθούν κβαντικές πληροφορίες, τότε ένας παίκτης μπορεί (θεωρητικά) πάντα να εξαπατήσει ανεξάρτητα από το πρωτόκολλο που χρησιμοποιείται. Αναφέρουμε τον όρο «θεωρητικά», επειδή η εξαπάτηση μπορεί να απαιτεί μια μη εφικτή ποσότητα υπολογιστικών πόρων. Σύμφωνα με τυπικές υπολογιστικές υποθέσεις, η ρίψη νομίσματος μπορεί να επιτευχθεί με την κλασική επικοινωνία.

Αυτό που χαρακτηρίζει ένα πρωτόκολλο ρίψης νομίσματος είναι η μεροληψία (bias) του, δηλαδή ένας αριθμός μεταξύ 0 και 1/2. Η μεροληψία ενός πρωτοκόλλου αποτυπώνει την πιθανότητα επιτυχίας ενός παντοδύναμου παίκτη που εξαπατά χρησιμοποιώντας την καλύτερη δυνατή στρατηγική. Ένα πρωτόκολλο με μεροληψία 0 σημαίνει ότι κανένας παίκτης δεν μπορεί να εξαπατήσει. Ένα πρωτόκολλο με μεροληψία 1/2 σημαίνει ότι τουλάχιστον ένας παίκτης μπορεί πάντα να πετύχει στην εξαπάτηση. Προφανώς, όσο μικρότερη είναι η μεροληψία, τόσο καλύτερο είναι το πρωτόκολλο. Όταν η επικοινωνία πραγματοποιείται μέσω ενός κβαντικού καναλιού, έχει αποδειχθεί ότι ακόμη και το καλύτερο δυνατό πρωτόκολλο δεν μπορεί να έχει μεροληψία μικρότερη από $1/\sqrt{2} - 1/2 \approx 0.2071$ [23].

Εάν εξετάσουμε την περίπτωση όπου κάθε παίκτης γνωρίζει το bit που προτιμά ο άλλος, τότε το πρόβλημα ρίψης νομίσματος που κάνει αυτή την πρόσθετη υπόθεση αποτελεί την ασθενέστερη παραλλαγή του που ονομάζεται αδύναμη ρίψη νομίσματος (Weak Coin Flipping – WCF) (βλέπε Ορισμό 14.1). Στην περίπτωση των κλασικών καναλιών, αυτή η επιπλέον υπόθεση δεν αποφέρει καμία βελτίωση. Από την άλλη πλευρά, έχει αποδειχθεί ότι υπάρχουν πρωτόκολλα WCF με αυθαίρετα μικρή μεροληψία [24]. Ωστόσο, το πιο γνωστό πρωτόκολλο WCF έχει μεροληψία $1/6 \approx 0.1667$ [25].

Ορισμός 14.1 (Ρίψη Νομίσματος). Στην κρυπτογραφία, η ρίψη νομίσματος ορίζεται ως το πρόβλημα όπου δύο αμοιβαία δύσπιστοι και απομακρυσμένοι παίκτες θέλουν να συμφωνήσουν σε ένα τυχαίο bit χωρίς να βασίζονται σε καμία τρίτη οντότητα [19]. Το πρόβλημα της ρίψης νομίσματος χωρίζεται σε δύο υποκατηγορίες:

- **Ισχυρή Ρίψη Νομίσματος:** Στην κβαντική κρυπτογραφία, η ισχυρή ρίψη νομίσματος (Strong Coin Flipping – SCF) ορίζεται ως ένα πρόβλημα ρίψης νομίσματος όπου ο κάθε παίκτης αγνοεί την προτίμηση του άλλου [26].
- **Αδύναμη Ρίψη Νομίσματος:** Στην κβαντική κρυπτογραφία, η αδύναμη ρίψη νομίσματος (Weak Coin Flipping – WCF) ορίζεται ως ένα πρόβλημα ρίψης νομίσματος όπου κάθε παίκτης γνωρίζει την προτίμηση του άλλου [27].

Αν και η κβαντική ρίψη νομίσματος προσφέρει σαφή πλεονεκτήματα σε σχέση με την κλασική ρίψη νομίσματος στη θεωρία, η επίτευξή του στην πράξη έχει αποδειχθεί ότι δεν είναι εύκολη υπόθεση [22]. Στις υποενότητες που ακολουθούν παρατίθενται δύο διαφορετικές παραλλαγές για την κβαντική ρίψη νομίσματος.

14.5.1 Ρίψη Νομίσματος με Χρήση Συζευγμένης Κωδικοποίησης

Η κβαντική ρίψη νομίσματος, όπως και άλλοι τύποι κβαντικής κρυπτογραφίας, επικοινωνούν πληροφορίες μέσω της μετάδοσης qubits. Ο παίκτης που συμμετέχει στο πρωτόκολλο δεν γνωρίζει τις πληροφορίες στο qubit μέχρι να εκτελέσει μια μέτρηση. Οι πληροφορίες για κάθε qubit αποθηκεύονται και μεταφέρονται από ένα μόνο φωτόνιο. Μόλις ο παραλήπτης μετρήσει το φωτόνιο, αυτό μεταβάλλεται και δεν θα παράγει την ίδια έξοδο εάν μετρηθεί ξανά. Δεδομένου ότι ένα φωτόνιο μπορεί να διαβαστεί με τον ίδιο τρόπο μόνο μία φορά, η προσπάθεια μιας τρίτης οντότητας να υποκλέψει το μήνυμα είναι εύκολα ανιχνεύσιμη.

Η κβαντική ρίψη νομίσματος πραγματοποιείται όταν δημιουργούνται τυχαία qubits μεταξύ των δύο παικτών που δεν εμπιστεύονται ο ένας τον άλλον επειδή και οι δύο θέλουν να κερδίσουν την ρίψη του νομίσματος, κάτι που θα μπορούσε να τους οδηγήσει στο να εξαπατήσουν ο ένας τον άλλον με διάφορους τρόπους [22]. Η ουσία της ρίψης νομίσματος λαμβάνει χώρα όταν οι δύο παίκτες εκτελούν μια σειρά εντολών μέσω ενός καναλιού επικοινωνίας που έχει ως αποτέλεσμα μια έξοδο.

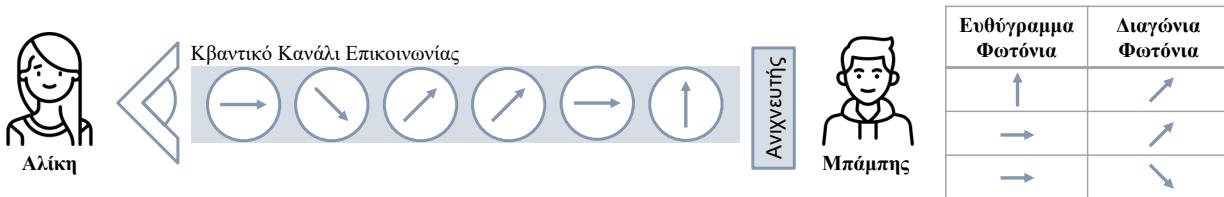
Ένα βασικό πρωτόκολλο [22] της κβαντικής ρίψης νομίσματος περιλαμβάνει δύο παίκτες, την Αλίκη και τον Μπάμπη, οι οποίοι ακολουθούν τα ακόλουθα βήματα κατά την εκτέλεση του πρωτοκόλλου:

1. Η Αλίκη στέλνει στον Μπάμπη έναν καθορισμένο αριθμό παλμών φωτονίων K στις κβαντικές καταστάσεις $|\phi_{\alpha_i c_i}\rangle$. Καθένας από αυτούς τους παλμούς φωτονίων προετοιμάζεται ανεξάρτητα ακολουθώντας μια τυχαία επιλογή από την Αλίκη, τόσο της βάσης α_i , όσο και του bit c_i , όπου $i = 1, 2, 3, \dots, K$.
2. Στη συνέχεια, ο Μπάμπης μετράει τους παλμούς φωτονίων που έστειλε η Αλίκη, προσδιορίζοντας τυχαία μια βάση β_i . Ο Μπάμπης καταγράφει αυτά τα φωτόνια και στη συνέχεια αναφέρει το πρώτο επιτυχώς καταμετρημένο φωτόνιο j στην Αλίκη μαζί με ένα τυχαίο bit b .
3. Η Αλίκη αποκαλύπτει τη βάση και το bit που χρησιμοποίησε για την βάση που της έδωσε ο Μπάμπης. Εάν οι δύο βάσεις και τα bit ταιριάζουν, τότε και τα δύο συμμετέχοντες είναι πραγματικοί (δηλ., αληθινοί) και μπορούν να ανταλλάξουν πληροφορίες. Αν το bit που αναφέρθηκε από τον Μπάμπη είναι διαφορετικό από αυτό της Αλίκης, τότε κάποιος δεν είναι αληθινός.

Μια πιο γενικευμένη εκδοχή του παραπάνω πρωτοκόλλου είναι η εξής [6]:

1. Η Αλίκη επιλέγει πρώτα μια τυχαία βάση (όπως η διαγώνιος) και μια ακολουθία τυχαίων qubits. Στη συνέχεια, η Αλίκη κωδικοποιεί τα επιλεγμένα qubits της ως μια ακολουθία φωτονίων που ακολουθούν την επιλεγμένη βάση, και στέλνει αυτά τα qubits ως μια ακολουθία πολωμένων φωτονίων στον Μπάμπη μέσω του καναλιού επικοινωνίας.
2. Ο Μπάμπης επιλέγει τυχαία μια ακολουθία βάσεων ανάγνωσης για κάθε μεμονωμένο φωτόνιο. Στη συνέχεια διαβάζει τα φωτόνια και καταγράφει τα αποτελέσματα σε δύο πίνακες (βλέπε Σχήμα 14.2). Ο ένας πίνακας είναι των ευθύγραμμων (οριζόντιων ή κατακόρυφων) λαμβανόμενων φωτονίων και ο άλλος πίνακας είναι των διαγώνιων λαμβανόμενων φωτονίων. Ο Μπάμπης μπορεί να έχει κενά στους πίνακες του λόγω απωλειών στους ανιχνευτές του ή στα κανάλια μετάδοσης. Με βάση αυτούς τους πίνακες, ο Μπάμπης μαντεύει ποια βάση χρησιμοποίησε η Αλίκη και την ανακοινώνει στην Αλίκη. Αν μάντεψε σωστά, κερδίζει και αν όχι, τότε χάνει.
3. Η Αλίκη αναφέρει αν κέρδισε ή όχι, ανακοινώνοντας ποια βάση χρησιμοποίησε στην επικοινωνία με τον Μπάμπη. Στη συνέχεια, η Αλίκη επιβεβαιώνει τις πληροφορίες στέλνοντας στον Μπάμπη ολόκληρη την αρχική της ακολουθία από qubit, που χρησιμοποίησε στο βήμα 1.

4. Ο Μπάμπης συγκρίνει την ακολουθία της Αλίκης με τους πίνακές του για να επιβεβαιώσει ότι δεν έγινε προσπάθεια εξαπάτησης από την πλευρά της Αλίκης. Οι πίνακες θα πρέπει να αντιστοιχούν στη βάση της Αλίκης και δεν θα πρέπει να υπάρχει συσχέτιση του ενός με τον άλλο πίνακα.



Σχήμα 14.2: Επίδειξη του πρωτοκόλλου της κβαντικής ρίψης νομίσματος.

Παραδοχές: Για να λειτουργήσει σωστά το παραπάνω πρωτόκολλο, υπάρχουν μερικές παραδοχές που πρέπει να γίνουν. Το πρώτη παραδοχή είναι ότι η Αλίκη μπορεί να δημιουργήσει κάθε κατάσταση ανεξάρτητα από τον Μπάμπη, και με ίσες πιθανότητες. Η δεύτερη παραδοχή είναι ότι για το πρώτο bit που μετρά επιτυχώς ο Μπάμπης, η βάση και το bit του είναι τυχαία και εντελώς ανεξάρτητα από την Αλίκη. Η τρίτη και τελευταία παραδοχή είναι ότι όταν ο Μπάμπης μετρά μια κατάσταση, έχει μια ομοιόμορφη πιθανότητα να μετρήσει κάθε κατάσταση και καμία κατάσταση δεν είναι πιο εύκολη να εντοπιστεί από άλλες. Αυτή η τελευταία παραδοχή είναι ιδιαίτερα σημαντική γιατί αν η Αλίκη γνώριζε την αδυναμία του Μπάμπη να μετρήσει ορισμένες καταστάσεις, θα μπορούσε να το χρησιμοποιήσει προς όφελός της [22].

Διαδικασίες εξαπάτησης: Το βασικό ζήτημα με την ρίψη νομίσματος είναι ότι λαμβάνει χώρα μεταξύ δύο μερών που δεν έχουν εμπιστοσύνη. Αυτά τα δύο μέρη επικοινωνούν μέσω του καναλιού επικοινωνίας σε κάποια απόσταση μεταξύ τους και πρέπει να συμφωνήσουν σε έναν νικητή ή σε έναν χαμένο, με τον καθένα να έχει 50% πιθανότητες να κερδίσει. Ωστόσο, δεδομένου ότι δεν έχουν εμπιστοσύνη ο ένας στον άλλον, είναι πιθανό να συμβεί κάποια εξαπάτηση. Η εξαπάτηση μπορεί να συμβεί με διάφορους τρόπους, όπως ο ισχυρισμός ότι έχασαν μέρος του μηνύματος όταν δεν τους αρέσει το αποτέλεσμα ή αύξηση του μέσου αριθμού φωτονίων που περιέχονται σε καθέναν από τους παλμούς. Για να μπορέσει να εξαπατήσει ο Μπάμπης, θα έπρεπε να είναι σε θέση να μαντέψει τη βάση της Αλίκης με πιθανότητα μεγαλύτερη από 1/2. Για να το επιτύχει αυτό, ο Μπάμπης θα έπρεπε να είναι σε θέση να προσδιορίσει μια ακολουθία φωτονίων τυχαία πολωμένων σε μια βάση από μια ακολουθία φωτονίων που πολώνονται σε άλλη βάση [6]. Η Αλίκη, από την άλλη, θα μπορούσε να εξαπατήσει με μερικούς διαφορετικούς τρόπους, αλλά πρέπει να είναι προσεκτική γιατί ο Μπάμπης θα μπορούσε εύκολα να το εντοπίσει. Όταν ο Μπάμπης στέλνει μια σωστή μαντεψία στην Αλίκη, η Αλίκη θα μπορούσε να πείσει τον Μπάμπη ότι τα φωτόνια της είναι πραγματικά πολωμένα, δηλαδή το αντίθετο από τη σωστή μαντεψία του Μπάμπη [6]. Επίσης, η Αλίκη θα μπορούσε να στείλει στον Μπάμπη μια διαφορετική αρχική ακολουθία από αυτή που χρησιμοποίησε στην πραγματικότητα για να νικήσει και πάλι τον Μπάμπη [6].

14.5.2 Το Πρωτόκολλο Dip Dip Boom

Το πρωτόκολλο Dip Dip Boom (DDB) [25] αποτελεί μια κβαντική έκδοση του παιχνιδιού που περιγράφεται παρακάτω. Έστω ότι έχουμε μια λίστα με αριθμούς p_i , ο καθένας μεταξύ 0 και 1. Οι παίκτες, η Αλίκη και ο Μπάμπης, παίζουν εναλλάξ και λένε “Dip” ή “Boom” με πιθανότητα p_i στο γύρο i . Ο παίκτης που λέει “Boom” κερδίζει. Προφανώς, ένας παίκτης που εξαπατά μπορεί απλά να πει “Boom” και να κερδίσει καθώς δεν υπάρχει ανταμοιβή για μεγαλύτερης διάρκειας παιχνίδια. Επιπλέον, θεωρούμε ότι τα παιχνίδια τερματίζονται για μερικά (μεγάλα) i , ας πούμε n , και ορίζουμε ότι $p_i = 1$. Με δεδομένο ότι βρισκόμαστε στο γύρο i , και $P_A(i)$ και $P_B(i)$ αντιπροσωπεύουν την πιθανότητα, αντίστοιχα, η Αλίκη και ο Μπάμπης να κερδίσουν. Έστω $P_U(i)$ επίσης η πιθανότητα το παιχνίδι να μην τερματίσει.

Στην κβαντική έκδοση αυτού του παιχνιδιού, έστω \mathbb{A}, \mathbb{B} ένας τρισδιάστατος χώρος Hilbert που εκτείνεται

μεταξύ $|A\rangle, |B\rangle, |U\rangle$, και έστω \mathbb{M} ένας δισδιάστατος χώρος Hilbert που εκτείνεται μεταξύ $|DIP\rangle, |BOOM\rangle$. Οπότε, τα βήματα του κβαντικού πρωτοκόλλου Dip Dip Boom διαμορφώνονται ως εξής:

1. **Αρχικοποίηση:** Η Αλίκη κατέχει τους καταχωρητές $\mathbb{A} \otimes \mathbb{M}$ και αρχικοποιεί την κατάσταση σε $|U\rangle \otimes |DIP\rangle$. Ο Μπάμπης κατέχει τον καταχωρητή \mathbb{B} και τον αρχικοποιεί στην κατάσταση $|U\rangle$.
2. **Επανάληψη:** Για $i = 1$ έως n εκτελούνται τα ακόλουθα. Για περιττό i ορίζουμε ότι $X = A$ (για την Αλίκη) και $Y = B$ (για τον Μπάμπη), ενώ για άρτιο i ορίζουμε ότι $X = B$ και $Y = A$.
 - Ο X υλοποιεί τη λειτουργία: $R_i = \text{Rot}(|U\rangle \otimes |DIP\rangle, |X\rangle \otimes |BOOM\rangle, p_i)$.
 - Ο X στέλνει τον καταχωρητή μηνυμάτων \mathbb{M} στον Y .
 - Ο Y υλοποιεί τη λειτουργία: $\tilde{R}_i = \text{Rot}\left(|U\rangle \otimes |BOOM\rangle, |X\rangle \otimes |DIP\rangle, \frac{p_i P_U(i-1)}{P_X(i)}\right)$.
 - Ο Y μετρά τον καταχωρητή μηνυμάτων σύμφωνα με την βάση υπολογισμού. Εάν το αποτέλεσμα είναι $BOOM$ τότε ο Y διακόπτει και ανακηρύσσει τον εαυτό του νικητή.
3. **Μέτρηση:** Η Αλίκη και ο Μπάμπης μετρούν και οι δύο τον τοπικό τους καταχωρητή \mathbb{A} και \mathbb{B} , αντίστοιχα. Εάν το αποτέλεσμα είναι U τότε δηλώνουν νικητές. Αν το αποτέλεσμα είναι A τότε η Αλίκη είναι η νικήτρια, και εάν είναι B τότε ο Μπάμπης είναι ο νικητής.

14.6 Κβαντική Δέσμευση

Εκτός από την κβαντική ρίψη νομίσματος, τα πρωτόκολλα κβαντικής δέσμευσης (quantum commitment) εφαρμόζονται όταν εμπλέκονται δύσπιστα μέρη. Ένα σχήμα δέσμευσης επιτρέπει σε ένα συμβαλλόμενο μέρος, την Αλίκη, να καθορίσει μια συγκεκριμένη τιμή (να «δεσμευτεί») με τέτοιο τρόπο ώστε η Αλίκη να μην μπορεί να αλλάξει αυτήν την τιμή, ενώ ταυτόχρονα διασφαλίζει ότι ο παραλήπτης, ο Μπάμπης, δεν μπορεί να μάθει τίποτα για αυτήν την τιμή μέχρι να την αποκαλύψει η Αλίκη. Τέτοια σχήματα δέσμευσης χρησιμοποιούνται συνήθως σε κρυπτογραφικά πρωτόκολλα, όπως η κβαντική ρίψη νομίσματος, οι αποδείξεις μηδενικής γνώσης, οι ασφαλείς υπολογισμοί πολλαπλών οντοτήτων και η μη-συνειδητή μεταφορά.

Σε κβαντικό επίπεδο, η κβαντική δέσμευση είναι ιδιαίτερα χρήσιμη, και όπως έδειξαν οι Crépeau και Kilian, κάνοντας χρήση μιας δέσμευσης και ενός κβαντικού καναλιού, μπορεί κανείς να κατασκευάσει ένα άνευ όρων ασφαλές πρωτόκολλο για την εκτέλεση της λεγόμενης μη-συνειδητής μεταφοράς [28]. Η μη-συνειδητή μεταφορά, από την άλλη πλευρά, έχει αποδειχθεί από τον Kilian [29] ότι επιτρέπει την υλοποίηση σχεδόν οποιουδήποτε κατανεμημένου υπολογισμού με ασφαλή τρόπο (τους αποκαλούμενους ασφαλείς υπολογισμούς πολλαπλών οντοτήτων). Διαστυχώς, τα πρώιμα πρωτόκολλα κβαντικής δέσμευσης [30] αποδείχθηκαν ελαττωματικά. Στην πραγματικότητα, ο Mayers αργότερα έδειξε ότι η (άνευ όρων ασφαλής) κβαντική δέσμευση είναι αδύνατη, γιατί ένας εισβολέας χωρίς υπολογιστικούς περιορισμούς μπορεί να σπάσει οποιοδήποτε πρωτόκολλο κβαντικής δέσμευσης [31].

Ωστόσο, τα αποτελέσματα του Mayers δεν αποκλείουν τη δυνατότητα κατασκευής πρωτοκόλλων κβαντικής δέσμευσης (και επομένως πρωτοκόλλων ασφαλών υπολογισμών πολλαπλών οντοτήτων) κάτω από παραδοχές που είναι πολύ πιο αδύναμες από τις παραδοχές που απαιτούνται για πρωτόκολλα δέσμευσης που δεν χρησιμοποιούν κβαντική επικοινωνία. Το οριθμητημένο μοντέλο κβαντικής αποθήκευσης, που περιγράφεται στην επόμενη ενότητα, αποτελεί ένα παράδειγμα στο οποίο η κβαντική επικοινωνία μπορεί να χρησιμοποιηθεί για την κατασκευή πρωτοκόλλων δέσμευσης. Σχετικά πρόσφατα τον Νοέμβριο του 2013, προτάθηκε επίσης ένα νέο σχήμα κβαντικής δέσμευσης [32] που παρέχει άνευ όρων ασφάλεια αξιοποιώντας την κβαντική θεωρία και τη σχετικότητα, και για το οποίο μάλιστα έχει αποδειχθεί με επιτυχία η ισχύς του σε παγκόσμιο επίπεδο. Ακόμη πιο πρόσφατα, ο Wang et al. [33], πρότεινε ένα άλλο σχήμα δέσμευσης στο οποίο η «άνευ όρων απόκρυψη» (unconditional hiding) είναι τέλεια. Τέλος, δείχτηκε ότι οι φυσικές μη-κλωνοποιήσιμες συναρτήσεις (physical unclonable functions) [34] μπορούν επίσης να αξιοποιηθούν για την κατασκευή κρυπτογραφικών δεσμεύσεων.

14.7 Οριοθετημένο και Θορυβώδες Μοντέλο Κβαντικής Αποθήκευσης

Μια επιπλέον δυνατότητα κατασκευής πρωτοκόλλων άνευ όρων ασφαλούς κβαντικής δέσμευσης και κβαντικής μη-συνειδητής μεταφοράς είναι η χρήση του οριοθετημένου μοντέλου κβαντικής αποθήκευσης (Bounded Quantum Storage Model – BQSM) [35]. Σε αυτό το μοντέλο, θεωρείται ότι η ποσότητα κβαντικών δεδομένων που μπορεί να αποθηκεύσει ένας αντίπαλος περιορίζεται από μια γνωστή σταθερά Q . Ωστόσο, δεν επιβάλλεται όριο στην ποσότητα των κλασικών (δηλ., μη κβαντικών) δεδομένων που μπορεί να αποθηκεύσει ο αντίπαλος.

Η βασική ιδέα για το πως μπορεί να χρησιμοποιηθεί το BQSM για την κατασκευή πρωτοκόλλων δέσμευσης και μη-συνειδητής μεταφοράς είναι η εξής: οι οντότητες του πρωτοκόλλου ανταλλάσσονται περισσότερα από Q κβαντικά bit (qubits). Δεδομένου ότι ακόμη και μια ανέντιμη οντότητα δεν μπορεί να αποθηκεύσει όλες αυτές τις πληροφορίες (η κβαντική μνήμη του αντίπαλου περιορίζεται σε Q qubits), ένα μεγάλο μέρος των δεδομένων θα πρέπει είτε να μετρηθούν είτε να απορριφθούν. Ο εξαναγκασμός των ανέντιμων οντοτήτων να μετρήσουν ένα μεγάλο μέρος των δεδομένων επιτρέπει στο πρωτόκολλο να παρακάμψει το αποτέλεσμα αδυναμίας (impossibility result), με αποτέλεσμα τα πρωτόκολλα δέσμευσης και μη-συνειδητής μεταφοράς να μπορούν πλέον να εφαρμοστούν [31].

Το πρωτόκολλο BQSM, που προτάθηκε από τον DamgÅrd *et al.* [35], δεν προϋποθέτει ότι οι τίμιοι συμμετέχοντες στο πρωτόκολλο αποθηκεύουν οποιαδήποτε κβαντική πληροφορία, και οι τεχνικές απαιτήσεις είναι παρόμοιες με εκείνες για τα πρωτόκολλα κβαντικής διανομής κλειδιών. Επομένως, αυτά τα πρωτόκολλα μπορούν, τουλάχιστον καταρχήν, να υλοποιηθούν με τη σημερινή τεχνολογία. Ένα επιπλέον πλεονέκτημα του BQSM είναι η παραδοχή ότι η κβαντική μνήμη του αντίπαλου είναι περιορισμένη, κάτιον οποίο είναι αρκετά ρεαλιστικό. Με τη σημερινή τεχνολογία, η αξιόπιστη αποθήκευση ακόμη και ενός qubit για αρκετά μεγάλο χρονικό διάστημα είναι δύσκολη. Το τι σημαίνει «αρκετά μεγάλο χρονικό διάστημα» εξαρτάται από τις λεπτομέρειες του πρωτοκόλλου. Με την εισαγωγή μιας τεχνητής παύσης στο πρωτόκολλο, ο χρόνος κατά τον οποίο χρειάζεται ο αντίπαλος για να αποθηκεύσει κβαντικά δεδομένα μπορεί να γίνει αυθαίρετα μεγάλος.

Μια επέκταση του BQSM αποτελεί το θορυβώδες μοντέλο κβαντικής αποθήκευσης (Noisy Quantum Storage Model – NQSM) που προτάθηκε από τους Wehner, Schaffner και Terhal [36]. Σε αυτό το μοντέλο, αντί να εξετάζεται το ανώτερο όριο στο φυσικό μέγεθος της κβαντικής μνήμης του αντίπαλου, επιτρέπεται σε έναν αντίπαλο να χρησιμοποιεί ατέλεις κβαντικές συσκευές αποθήκευσης αυθαίρετου μεγέθους. Το επίπεδο ατέλειας των κβαντικών αυτών συσκευών αποθήκευσης μοντελοποιείται με την χρήση θορυβώδων κβαντικών καναλιών (noisy quantum channels). Για αρκετά υψηλά επίπεδα θορύβου, μπορούν να επιτευχθούν τα ίδια πρωτογενή στοιχεία όπως στο BQSM [37] και το BQSM ουσιαστικά αποτελεί μια ειδική περίπτωση του θορυβώδους κβαντικού μοντέλου αποθήκευσης.

Σε ένα κλασικό περιβάλλον, παρόμοια αποτελέσματα μπορούν να επιτευχθούν όταν υποθέσουμε ένα όριο στην ποσότητα των κλασικών (μη κβαντικών) δεδομένων που μπορεί να αποθηκεύσει ο αντίπαλος [38]. Ωστόσο, έχει αποδειχθεί ότι σε αυτό το μοντέλο και οι τίμιες οντότητες πρέπει να χρησιμοποιήσουν μεγάλη ποσότητα μνήμης (δηλ., την τετραγωνική ρίζα της δεσμευμένης μνήμης του αντίπαλου) [39]. Αυτό καθιστά αυτά τα πρωτόκολλα μη πρακτικά για ρεαλιστικά όρια μνήμης.

14.8 Κβαντική Κρυπτογραφία βάσει Θέσης

Ο στόχος της κβαντικής κρυπτογραφίας βάσει θέσης (position-based quantum cryptography) είναι να χρησιμοποιήσει τη γεωγραφική θέση ενός παίκτη ως (μόνο) διαπιστευτήριό του. Για παράδειγμα, κάποιος θέλει να στείλει ένα μήνυμα σε έναν παίκτη σε μια καθορισμένη θέση με την εγγύηση ότι μπορεί να διαβαστεί μόνο εάν ο παραλήπτης βρίσκεται στη συγκεκριμένη θέση. Κατά την διαδικασία επαλήθευσης θέσης, μια παίκτρια, η Αλίκη, θέλει να πείσει τους (τίμιους) επαληθευτές (verifiers) ότι βρίσκεται σε ένα συγκεκριμένο σημείο. Όπως έχει αποδειχθεί από τον Chandran *et al.* [40], η επαλήθευση θέσης με χρήση κλασικών πρωτοκόλλων είναι αδύνατη ενάντια σε αντίπαλους που συνεννοούνται (οι οποίοι ελέγχουν όλες τις θέσεις εκτός από τη

θέση που διεκδικεί ο αποδεικνύων (prover)). Ωστόσο, κάτω από διάφορους περιορισμούς για τους αντιπάλους, τέτοια σχήματα είναι εφικτά.

Τα πρώτα κβαντικά σχήματα βάσει θέσης, με την ονομασία «κβαντική σήμανση» (quantum tagging), προτάθηκαν από τον Kent και κατοχυρώθηκαν ως πατέντα το 2006 [41]. Η έννοια της χρήσης κβαντικών φαινομένων (effects) για την επαλήθευση τοποθεσίας εμφανίστηκε για πρώτη φορά στην επιστημονική βιβλιογραφία το 2010 [42, 43]. Ωστόσο, ο Buhrman *et al.* [44] έδειξε ότι χρησιμοποιώντας ένα τεράστιο ποσό κβαντικής εμπλοκής (entanglement), οι αντίπαλοι που συνεννοούνται είναι πάντα σε θέση να έλεγχουν εάν οι επαληθευτές βρίσκονταν στη διεκδικούμενη θέση. Ωστόσο, αυτό το αποτέλεσμα δεν αποκλείει τη δυνατότητα ύπαρξης πρακτικών σχημάτων στο μοντέλο οριοθετημένης ή θορυβώδους κβαντικής αποθήκευσης (βλέπε Ενότητα 14.7).

Βιβλιογραφία

- [1] Nicolas Gisin, Grégoire Ribordy, Wolfgang Tittel, and Hugo Zbinden. “Quantum Cryptography”. In: *Reviews of Modern Physics* 74 (1 Mar. 2002), pp. 145–195. doi: 10.1103/RevModPhys.74.145.
- [2] Stefano Pirandola et al. “Advances in Quantum Cryptography”. In: *Advances in Optics and Photonics* 12.4 (Dec. 2020), pp. 1012–1236. doi: 10.1364/AOP.361502.
- [3] William K. Wootters and Wojciech H. Zurek. “A Single Quantum Cannot Be Cloned”. In: *Nature* 299.5886 (1982), pp. 802–803. doi: 10.1038/299802a0.
- [4] Charles H. Bennett, François Bessette, Gilles Brassard, Louis Salvail, and John Smolin. “Experimental Quantum Cryptography”. In: *Journal of Cryptology* 5.1 (1992), pp. 3–28. doi: 10.1007/BF00191318.
- [5] Stephen Wiesner. “Conjugate Coding”. In: *SIGACT News* 15.1 (Jan. 1983), pp. 78–88. ISSN: 0163-5700. doi: 10.1145/1008908.1008920.
- [6] Charles H. Bennett and Gilles Brassard. “Quantum Cryptography: Public Key Distribution and Coin Tossing”. In: *Theoretical Computer Science* 560 (2014). Theoretical Aspects of Quantum Cryptography – Celebrating 30 Years of BB84, pp. 7–11. ISSN: 0304-3975. doi: 10.1016/j.tcs.2014.05.025.
- [7] Artur K. Ekert. “Quantum Cryptography based on Bell’s Theorem”. In: *Physical Review Letters* 67 (6 Aug. 1991), pp. 661–663. doi: 10.1103/PhysRevLett.67.661.
- [8] Dominic Mayers and Andrew Yao. “Quantum Cryptography with Imperfect Apparatus”. In: *39th Annual Symposium on Foundations of Computer Science*. IEEE, 1998, pp. 503–509. doi: 10.1109/SFCS.1998.743501.
- [9] Christian Kollmitzer and Mario Pivk. *Applied Quantum Cryptography*. Vol. 797. Springer, 2010. ISBN: 978-3-642-04829-6. doi: 10.1007/978-3-642-04831-9.
- [10] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th. New York, NY, USA: Cambridge University Press, 2011. ISBN: 978-1-107-00217-3.
- [11] C. E. Shannon. “Communication Theory of Secrecy Systems”. In: *The Bell System Technical Journal* 28.4 (1949), pp. 656–715. doi: 10.1002/j.1538-7305.1949.tb00928.x.
- [12] Marco Tomamichel and Anthony Leverrier. “A Largely Self-Contained and Complete Security Proof for Quantum Key Distribution”. In: *Quantum* 1 (July 2017), p. 14. ISSN: 2521-327X. doi: 10.22331/q-2017-07-14-14. URL: <https://doi.org/10.22331/q-2017-07-14-14>.
- [13] J. S. Bell. “On the Einstein Podolsky Rosen Paradox”. In: *Physics Physique Fizika* 1 (3 Nov. 1964), pp. 195–200. doi: 10.1103/PhysicsPhysiqueFizika.1.195.

- [14] Adi Shamir. "How to Share a Secret". In: *Commun. ACM* 22.11 (Nov. 1979), pp. 612–613. ISSN: 0001-0782. doi: 10.1145/359168.359176.
- [15] George R. Blakley. "Safeguarding Cryptographic Keys". In: *International Workshop on Managing Requirements Knowledge*. Los Alamitos, CA, USA: IEEE, June 1979, p. 313. doi: 10.1109/AFIPS.1979.98.
- [16] Mark Hillery, Vladimír Bužek, and André Berthiaume. "Quantum Secret Sharing". In: *Physical Review A* 59 (3 Mar. 1999), pp. 1829–1834. doi: 10.1103/PhysRevA.59.1829.
- [17] Richard Cleve, Daniel Gottesman, and Hoi-Kwong Lo. "How to Share a Quantum Secret". In: *Phys. Rev. Lett.* 83 (3 July 1999), pp. 648–651. doi: 10.1103/PhysRevLett.83.648.
- [18] Daniel Gottesman. "Theory of Quantum Secret Sharing". In: *Phys. Rev. A* 61 (4 Mar. 2000), p. 042311. doi: 10.1103/PhysRevA.61.042311.
- [19] Manuel Blum. "Coin Flipping by Telephone a Protocol for Solving Impossible Problems". In: *SIGACT News* 15.1 (Jan. 1983), pp. 23–27. ISSN: 0163-5700. doi: 10.1145/1008908.1008911.
- [20] Oded Goldreich. *Foundations of Cryptography: Volume II Basic Applications*. Cambridge University Press, 2009. ISBN: 978-0-521-83084-3.
- [21] Michael Ben-Or and Avinatan Hassidim. "Fast Quantum Byzantine Agreement". In: *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*. STOC '05. Baltimore, MD, USA: Association for Computing Machinery, 2005, pp. 481–485. ISBN: 1581139608. doi: 10.1145/1060590.1060662.
- [22] Anna Pappa et al. "Experimental Plug and Play Quantum Coin Flipping". In: *Nature Communications* 5.1 (2014), pp. 1–8. doi: 10.1038/ncomms4717.
- [23] Andris Ambainis, Harry Buhrman, Yevgeniy Dodis, and Hein Rohrig. "Multiparty Quantum Coin Flipping". In: *Proceedings of the 19th IEEE Annual Conference on Computational Complexity*. 2004, pp. 250–259. doi: 10.1109/CCC.2004.1313848.
- [24] Dorit Aharonov, André Chailloux, Maor Ganz, Iordanis Kerenidis, and Loick Magnin. "A Simpler Proof of the Existence of Quantum Weak Coin Flipping with Arbitrarily Small Bias". In: *SIAM Journal on Computing* 45.3 (2016), pp. 633–679. doi: 10.1137/14096387X.
- [25] Carlos Mochon. "Large family of quantum weak coin-flipping protocols". In: *Physical Review A* 72 (2 Aug. 2005), p. 022341. doi: 10.1103/PhysRevA.72.022341.
- [26] Dorit Aharonov, Amnon Ta-Shma, Umesh V. Vazirani, and Andrew C. Yao. "Quantum Bit Escrow". In: *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*. STOC '00. Portland, Oregon, USA: Association for Computing Machinery, 2000, pp. 705–714. ISBN: 1581131844. doi: 10.1145/335305.335404.
- [27] Robert W. Spekkens and Terry Rudolph. "Quantum Protocol for Cheat-Sensitive Weak Coin Flipping". In: *Physical Review Letters* 89 (22 Nov. 2002), p. 227901. doi: 10.1103/PhysRevLett.89.227901.
- [28] Claude Crépeau and Joe Kilian. "Achieving Oblivious Transfer Using Weakened Security Assumptions". In: *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*. SFCS '88. USA: IEEE, 1988, pp. 42–52. ISBN: 0818608773. doi: 10.1109/SFCS.1988.21920.
- [29] Joe Kilian. "Founding Cryptography on Oblivious Transfer". In: *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*. STOC '88. Chicago, Illinois, USA: Association for Computing Machinery, 1988, pp. 20–31. ISBN: 0897912640. doi: 10.1145/62212.62215.

- [30] Gilles Brassard, Claude Crépeau, Richard Jozsa, and Denis Langlois. “A Quantum Bit Commitment Scheme Provably Unbreakable by Both Parties”. In: *Proceedings of IEEE 34th Annual Foundations of Computer Science*. 1993, pp. 362–371. doi: [10.1109/SFCS.1993.366851](https://doi.org/10.1109/SFCS.1993.366851).
- [31] Dominic Mayers. “Unconditionally Secure Quantum Bit Commitment is Impossible”. In: *Physical Review Letters* 78 (17 Apr. 1997), pp. 3414–3417. doi: [10.1103/PhysRevLett.78.3414](https://doi.org/10.1103/PhysRevLett.78.3414).
- [32] Tommaso Lunghi et al. “Experimental Bit Commitment Based on Quantum Communication and Special Relativity”. In: *Physical Review Letters* 111 (18 Nov. 2013), p. 180504. doi: [10.1103/PhysRevLett.111.180504](https://doi.org/10.1103/PhysRevLett.111.180504).
- [33] Ming-Qiang Wang, Xue Wang, and Tao Zhan. “Unconditionally Secure Multi-Party Quantum Commitment Scheme”. In: *Quantum Information Processing* 17.2 (2018), pp. 1–13.
- [34] Georgios M. Nikolopoulos. “Optical Scheme for Cryptographic Commitments with Physical Unclonable Keys”. In: *Optics Express* 27.20 (Sept. 2019), pp. 29367–29379. doi: [10.1364/OE.27.029367](https://doi.org/10.1364/OE.27.029367).
- [35] Ivan B. Damgård, Serge Fehr, Louis Salvail, and Christian Schaffner. “Cryptography in the Bounded-Quantum-Storage Model”. In: *SIAM Journal on Computing* 37.6 (2008), pp. 1865–1890. doi: [10.1137/060651343](https://doi.org/10.1137/060651343).
- [36] Stephanie Wehner, Christian Schaffner, and Barbara M. Terhal. “Cryptography from Noisy Storage”. In: *Physical Review Letters* 100 (22 June 2008), p. 220502. doi: [10.1103/PhysRevLett.100.220502](https://doi.org/10.1103/PhysRevLett.100.220502).
- [37] Robert Konig, Stephanie Wehner, and Jürg Wullschleger. “Unconditional Security From Noisy Quantum Storage”. In: *IEEE Transactions on Information Theory* 58.3 (2012), pp. 1962–1984. doi: [10.1109/TIT.2011.2177772](https://doi.org/10.1109/TIT.2011.2177772).
- [38] Christian Cachin, Claude Crépeau, and Julien Marcil. “Oblivious Transfer with a Memory-Bound Receiver”. In: *Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No. 98CB36280)*. IEEE, 1998, pp. 493–502. doi: [10.1109/SFCS.1998.743500](https://doi.org/10.1109/SFCS.1998.743500).
- [39] Stefan Dziembowski and Ueli Maurer. “On Generating the Initial Key in the Bounded-Storage Model”. In: *Advances in Cryptology - EUROCRYPT 2004*. Ed. by Christian Cachin and Jan L. Camenisch. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 126–137. ISBN: 978-3-540-24676-3. doi: [10.1007/978-3-540-24676-3_8](https://doi.org/10.1007/978-3-540-24676-3_8).
- [40] Nishanth Chandran, Vipul Goyal, Ryan Moriarty, and Rafail Ostrovsky. “Position Based Cryptography”. In: *Advances in Cryptology - CRYPTO 2009*. Ed. by Shai Halevi. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 391–407. ISBN: 978-3-642-03356-8. doi: [10.1007/978-3-642-03356-8_23](https://doi.org/10.1007/978-3-642-03356-8_23).
- [41] Adrian P. Kent, William J. Munro, Timothy P. Spiller, and Raymond G. Beausoleil. “Tagging Systems”. U.S. pat. 7075438B2. July 11, 2006.
- [42] Robert A. Malaney. “Location-Dependent Communications using Quantum Entanglement”. In: *Physical Review A* 81 (4 Apr. 2010), p. 042319. doi: [10.1103/PhysRevA.81.042319](https://doi.org/10.1103/PhysRevA.81.042319).
- [43] Adrian Kent, William J. Munro, and Timothy P. Spiller. “Quantum Tagging: Authenticating Location via Quantum Information and Relativistic Signaling Constraints”. In: *Phys. Rev. A* 84 (1 July 2011), p. 012326. doi: [10.1103/PhysRevA.84.012326](https://doi.org/10.1103/PhysRevA.84.012326).
- [44] Harry Buhrman et al. “Position-Based Quantum Cryptography: Impossibility and Constructions”. In: *SIAM Journal on Computing* 43.1 (2014), pp. 150–178. doi: [10.1137/130913687](https://doi.org/10.1137/130913687).

ΚΕΦΑΛΑΙΟ 15

ΜΕΤΑ-ΚΒΑΝΤΙΚΗ ΚΡΥΠΤΟΓΡΑΦΙΑ

Περίληψη

Η μετα-κβαντική (post-quantum) κρυπτογραφία αναφέρεται σε κρυπτογραφικούς αλγόριθμους που πιστεύεται ότι είναι ασφαλείς έναντι της υπολογιστικής ισχύος των κβαντικών υπολογιστών. Το πρόβλημα που αντιμετωπίζουν οι παραδοσιακοί αλγόριθμοι δημοσίου κλειδιού είναι ότι η ανθεκτικότητά τους βασίζεται σε δύσκολα μαθηματικά προβλήματα, όπως το πρόβλημα της παραγοντοποίησης και του διακριτού λογαρίθμου. Όλα αυτά τα προβλήματα μπορούν να επιλυθούν εύκολα με έναν ισχυρό κβαντικό υπολογιστή κάνοντας χρήση του κβαντικού αλγορίθμου του Shor. Ως αποτέλεσμα, τα παραδοσιακά κρυπτοσυστήματα δημοσίου κλειδιού μπορούν να παραβιαστούν αποτελεσματικά με τη χρήση των κβαντικών υπολογιστών. Στο κεφάλαιο αυτό παρουσιάζονται πέντε βασικές κατηγορίες κρυπτογραφικών συστημάτων ανθεκτικών σε κβαντικούς υπολογιστές (Ενότητα 15.3), καθώς και αλγόριθμοι για μια από αυτές τις κατηγορίες, αυτών που βασίζονται σε συναρτήσεις σύνοψης (Ενότητα 15.3.1). Επιπλέον, δίνονται κάποιες κατευθύνσεις για τη μετάβαση στην μετα-κβαντική εποχή (Ενότητα 15.4).

Προαπαιτούμενη γνώση: Κατανόηση των βασικών εννοιών της Κρυπτογραφίας που παρατίθενται στα εισαγωγικά κεφάλαια (Κεφάλαιο 1 έως 3) αυτού του βιβλίου.

15.1 Εισαγωγή

Η Κβαντική Υπολογιστική είναι ένας ανατρεπτικός και καινοτόμος τομέας της Κβαντικής Τεχνολογίας. Από τα μέσα της δεκαετίας του '90, οι επιστήμονες θεώρησαν ότι οι αλγόριθμοι κβαντικών υπολογιστών, δεδομένης της ύπαρξης ενός αρκετά ισχυρού κβαντικού υπολογιστή, μπορούν να αποτελέσουν σοβαρή απειλή για τα ευρέως χρησιμοποιούμενα κρυπτοσυστήματα δημοσίου κλειδιού, όπως αυτά που βασίζονται στον RSA και σε ECC, ή να αποδυναμώσουν τους τυποποιημένους αλγόριθμους συμμετρικής κρυπτογράφησης. Συγκεκριμένα, το 1994, ο Peter Shor απέδειξε ότι ένας αρκετά ισχυρός κβαντικός υπολογιστής μπορεί να λύσει τα μαθηματικά προβλήματα στα οποία βασίζονται τα πιο δημοφιλή συστήματα κρυπτογραφίας δημόσιου κλειδιού, όπως το RSA και το ECC, σε πολυωνυμικό χρόνο, καθώς και να μειώσει την ασφάλεια των τυποποιημένων

Γεώργιος Δροσάτος, Ιωάννης Μαυρίδης, Κωνσταντίνος Ράντος (2024). «Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας». Αθήνα: Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις. <https://doi.org/10.xxxx/978-618-85370-x-x>

 Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0

αλγορίθμων συμμετρικής κρυπτογράφησης [1].

Τον Οκτώβριο του 2019 [2], η Google ανακοίνωσε ότι επέτυχε την «κβαντική υπεροχή» (quantum supremacy), δηλαδή την ικανότητα ενός κβαντικού υπολογιστή να εκτελέσει μια εργασία που είναι αδύνατη για έναν κλασικό υπολογιστή, ανοίγοντας έτσι νέους ορίζοντες για την κβαντική τεχνολογία και δείχνοντας πως τα περισσότερα από τα σημερινά συστήματα κρυπτογραφίας δημόσιου κλειδιού που χρησιμοποιούνται για την προστασία των δεδομένων και των επικοινωνιών στο διαδίκτυο απειλούνται σημαντικά. Λαμβάνοντας υπόψη τις δυνατότητες της κβαντικής υπολογιστικής, το NIST ξεκίνησε το 2016 μια διαδικασία για την τυποποίηση ενός ή περισσότερων κρυπτογραφικών αλγορίθμων δημόσιου κλειδιού ανθεκτικών σε κβαντικές επιθέσεις, ζητώντας προτάσεις από κρυπτογράφους σε όλο τον κόσμο¹.

Σε αυτό το κεφάλαιο θα αναλύσουμε τα θέματα που σχετίζονται με τη Μετα-Κβαντική Κρυπτογραφία (Post-Quantum Cryptography – PQC) και τις λύσεις που έχουν προταθεί στο πλαίσιο της επίλυσης του παραπάνω προβλήματος. Πριν προχωρήσουμε, ωστόσο, είναι σημαντικό να γίνει κατανοητή η διάκριση μεταξύ Μετα-Κβαντικής Κρυπτογραφίας και Κβαντικής Κρυπτογραφίας, όπως αυτή αναλύθηκε στο προηγούμενο κεφάλαιο. Η Μετα-Κβαντική Κρυπτογραφία, αντικείμενο αυτού του κεφαλαίου, αφορά το σχεδιασμό κρυπτογραφικών λύσεων που μπορούν να χρησιμοποιηθούν και από τους σημερινούς (μη κβαντικούς) υπολογιστές και που πιστεύουμε ότι είναι ανθεκτικές τόσο στη συμβατική όσο και στην κβαντική κρυπτανάλυση. Από την άλλη πλευρά, η Κβαντική Κρυπτογραφία αφορά κρυπτογραφικές λύσεις που εκμεταλλεύονται την κβαντική φυσική για να παρέχουν ορισμένες υπηρεσίες ασφαλείας, όπως αυτές αναλύθηκαν στο Κεφάλαιο 14.

15.2 Μετα-Κβαντική Κρυπτογραφία

Η μετα-κβαντική κρυπτογραφία (Post-Quantum Cryptography) είναι ένας τομέας της κρυπτογραφίας που επικεντρώνεται στην ανάπτυξη κρυπτογραφικών συστημάτων τα οποία θεωρούνται ασφαλή απέναντι σε επιθέσεις από κβαντικούς υπολογιστές. Αυτός ο τομέας παρουσιάζει ιδιαίτερο ενδιαφέρον λόγω της μελέτης του Peter Shor [1], βάσει της οποίας, σε ένα περιβάλλον όπου οι επιτιθέμενοι διαθέτουν κβαντικούς υπολογιστές, τα παραδοσιακά κρυπτογραφικά συστήματα δημόσιου κλειδιού δεν είναι ασφαλή.

Η συμμετρική κρυπτογραφία επηρεάζεται επίσης από τις δυνατότητες των κβαντικών υπολογιστών, αν και σημαντικά λιγότερο. Για συστήματα που δεν βασίζονται σε μαθηματικές δομές, η βασικότερη επίπτωση προκύπτει από τον αλγόριθμο που προτάθηκε από τον Lov Grover το 1996 [3], και ο οποίος μειώνει στο μισό το επίπεδο ασφάλειας τέτοιων κρυπτοσυστημάτων. Αυτό σημαίνει ότι το σπάσιμο του AES-128 απαιτεί 2^{64} κβαντικές λειτουργίες, ενώ οι τρέχουσες επιθέσεις απαιτούν 2^{128} δοκιμές εξαντλητικής αναζήτησης. Αν και αυτή είναι μια μεγάλη αλλαγή, μπορεί να αντιμετωπιστεί πολύ εύκολα διπλασιάζοντας τα μεγέθη των κλειδιών, π.χ. με την ανάπτυξη του AES-256. Οι λειτουργίες που απαιτούνται στον αλγόριθμο του Grover είναι εγγενώς διαδοχικές, δηλαδή δεν μπορούν να γίνουν ταυτόχρονα ή παράλληλα, γεγονός που έχει κάνει ορισμένους να αμφιβάλλουν εάν και κατά πόσο 2^{64} κβαντικές πράξεις είναι εφικτές. Ωστόσο, επειδή το αντίμετρο που προτείνεται, δηλαδή η αλλαγή σε μεγαλύτερα μεγέθη κλειδιών, είναι πολύ φθηνό, συνιστάται γενικά η υιοθέτησή του.

Οι κβαντικοί υπολογιστές που έχουν δημιουργηθεί μέχρι τώρα δεν είναι αρκετά ισχυροί ώστε να αποτελούν απειλή για την τρέχουσα κρυπτογραφία. Ωστόσο, η ανάπτυξη νέων κρυπτογραφικών συστημάτων απαιτεί πολύ χρόνο και προσπάθεια, επομένως είναι σημαντικό να υπάρχουν επιλογές αντικατάστασής τους πολύ πριν από την εμφάνιση μεγάλων, ισχυρών κβαντικών συστημάτων.

Ιδιαίτερα ανησυχητικό είναι το γεγονός ότι κάθε κρυπτογραφημένο κείμενο που έχει υποκλαπεί ή θα υποκλαπεί από έναν επιτιθέμενο σήμερα μπορεί να αποκρυπτογραφηθεί μόλις ο επιτιθέμενος αποκτήσει πρόσβαση σε έναν ισχυρό κβαντικό υπολογιστή. Αυτή η κατάσταση ονομάζεται αναδρομική αποκρυπτογράφηση (Retrospective Decryption). Οι φορείς Προηγμένων Επίμονων Απειλών (Advanced Persistent Threats – APT), ιδίως αυτοί που λειτουργούν σε επίπεδο κρατικών απειλών, ενδέχεται να καταγράφουν μεγάλο τμήμα της κίνησης του διαδικτύου στα κέντρα δεδομένων τους, συμπεριλαμβανομένης της κρυπτογραφημένης κί-

¹<https://csrc.nist.gov/news/2016/public-key-post-quantum-cryptographic-algorithms>

νησης, λόγω του πιθανού ενδιαφέροντος που παρουσιάζει. Αυτό σημαίνει ότι όλα τα δεδομένα που προστατεύονται με τα τυπικά συστήματα δημόσιου κλειδιού σήμερα, θα πρέπει να θεωρηθούν παραβιασμένα στη μετα-κβαντική εποχή, μόλις δηλαδή δημιουργηθούν ισχυροί κβαντικοί υπολογιστές.

Αξίζει να σημειωθεί ότι έναντι αυτής της απειλής δεν υπάρχει τρόπος να προστατευτεί το σύστημα αναδρομικά, καθώς ένα αντίγραφο του κρυπτογραφημένου κειμένου βρίσκεται στα χέρια του φορέα απειλής. Αυτό σημαίνει ότι τα δεδομένα που πρέπει να παραμείνουν εμπιστευτικά και μετά την άφιξη των κβαντικών υπολογιστών πρέπει να κρυπτογραφούνται με εναλλακτικά μέσα, δηλαδή με αλγορίθμους οι οποίοι θεωρούνται ασφαλείς για χρήση στη μετα-κβαντική εποχή.

Οι ψηφιακές υπογραφές αναμένεται επίσης να επηρεαστούν από τη χρήση των κβαντικών υπολογιστών, καθώς η ασφάλειά τους βασίζεται σε προβλήματα θεωρίας αριθμών. Ωστόσο, υπάρχει τουλάχιστον μία οικογένεια που θα παραμείνει ασφαλής έναντι της ανάπτυξης κβαντικών υπολογιστών. Αυτή η οικογένεια περιλαμβάνει τα λεγόμενα συστήματα υπογραφών που βασίζονται σε συναρτήσεις σύνοψης (hash-based digital signatures) [4, 5]. Πρόκειται για έναν τύπο ψηφιακών υπογραφών στις οποίες οι συναρτήσεις σύνοψης αποτελούν κεντρικό δομικό στοιχείο. Θεωρούνται αποτελεσματικές και ευέλικτες και μπορούν να χρησιμοποιηθούν σε ποικίλες εφαρμογές, ενώ η ασφάλειά τους βασίζεται στις ιδιότητες ασφαλείας των υποκείμενων συναρτήσεων σύνοψης. Με τα μέχρι τώρα δεδομένα, φαίνεται πως οι κβαντικοί υπολογιστές δεν μπορούν να παραβιάσουν την ασφάλεια των συναρτήσεων σύνοψης και, επομένως, οι υπογραφές αυτού του τύπου αναμένεται να παραμείνουν ασφαλείς.

Ωστόσο, αυτές οι υπογραφές δεν θεωρούνται κατάλληλες για γενική χρήση. Έχουν ειδικές απαιτήσεις διαχείρισης των ιδιωτικών κλειδιών υπογραφής τα οποία θα χρησιμοποιούνται μόνο μια φορά (υπογραφές μιας χρήσης – αναλύονται στην Ενότητα 15.3.1.1). Συγκεκριμένα, ένα ιδιωτικό κλειδί υπογραφής δεν πρέπει να χρησιμοποιείται για την υπογραφή περισσότερων του ενός μηνυμάτων. Διαφορετικά, θα ήταν υπολογιστικά εφικτό για έναν επιτιθέμενο να δημιουργήσει πλαστές υπογραφές για οποιαδήποτε μηνύματα. Λόγω αυτών των ιδιαιτεροτήτων, συνιστώνται μόνο για συστήματα με συγκεκριμένα χαρακτηριστικά, όπως αυτά με μεγάλη διάρκεια ζωής, όπου θα είναι δύσκολη η αναβάθμιση σε ένα σύστημα υπογραφών που βασίζεται σε συναρτήσεις σύνοψης.

Τα κλειδιά των ψηφιακών υπογραφών μπορούν να αντικατασταθούν και τα παλιά κλειδιά μπορούν να ανακληθούν όταν ένα σύστημα υπογραφών είναι κατεστραμμένο. Ωστόσο, μέρος των προσπαθειών και εξελίξεων στον τομέα της κατασκευής κβαντικών υπολογιστών ενδέχεται να μη δημοσιοποιείται και έτσι είναι αρκετά πιθανό ο πρώτος πλήρως λειτουργικός μεγάλος κβαντικός υπολογιστής να μην ανακοινωθεί δημόσια, αλλά να αποτελέσει αντικείμενο εκμετάλλευσης κάποιας κυβερνητικής υπηρεσίας. Επομένως, το βέλτιστο χρονικό σημείο αλλαγής των κλειδιών υπογραφής μπορεί να μην καταστεί σαφές.

Το 2017, το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας των Ηνωμένων Πολιτειών (National Institute of Standards and Technology – NIST) δημοσίευσε ανοιχτή πρόσκληση για υποβολές προτάσεων για νέους αλγορίθμους δημοσίου κλειδιού για κρυπτογράφηση και υπογραφή που θα είναι ασφαλείς στην μετα-κβαντική εποχή. Αν και δεν θεωρείται επίσημα «διαγωνισμός» όπως ήταν οι αντίστοιχες ενέργειες για τους AES και SHA-3, η όλη διαδικασία επιλογής αντιμετωπίζεται σχεδόν με τον ίδιο τρόπο. Ως αποτέλεσμα, στον τρίτο γύρο της διαδικασίας επιλογής, τον Ιούλιο του 2020, επιλέχθηκε ένα σύνολο κατάλληλων για την μετα-κβαντική εποχή μηχανισμών.

15.3 Οικογένειες Μετα-Κβαντικών Αλγορίθμων

Οι συνηθισμένες, αν και άτυπες, δηλώσεις αποποίησης ευθυνών που συνοδεύουν τη χρήση κρυπτογραφικών αλγορίθμων ισχύουν και για τα μετα-κβαντικά συστήματα, καθώς δεν μπορεί να αποκλειστεί το ενδεχόμενο ύπαρξης ισχυρότερων επιθέσεων (κβαντικών ή μη) που δεν έχουν ακόμη ανακαλυφθεί και θα μπορούσαν να παραβιάσουν επιτυχώς και αυτά τα συστήματα. Ωστόσο, η έρευνα των τελευταίων 15-20 ετών έχει δημιουργήσει εμπιστοσύνη στους τομείς που καταγράφονται σε αυτό το κεφάλαιο, οι οποίοι θεωρούνται ότι οδηγούν σε ασφαλή συστήματα σε έναν μετα-κβαντικό κόσμο. Σε αυτό το κεφάλαιο, εστιάζουμε στη μελέτη των αλγο-

ρίθμων που βασίζονται σε συνόψεις, παρέχοντας ωστόσο μια σύντομη αναφορά στις άλλες κατηγορίες.

15.3.1 Κρυπτογραφία Βασισμένη σε Συναρτήσεις Σύνοψης

Οι συναρτήσεις σύνοψης είναι ένα από τα πιο ευρέως διαδεδομένα κρυπτογραφικά εργαλεία, με εφαρμογές που κυμαίνονται από σύνοψη συνθηματικού έως αθροίσματα ελέγχου αρχείων, και χρησιμοποιούνται ουσιαστικά σε μια πληθώρα κρυπτοσυστημάτων. Επιπλέον, πρακτικά χρησιμοποιούνται σε όλα τα συστήματα υπογραφών για το χειρισμό μηνυμάτων αυθαίρετου μήκους, όπως συμβαίνει με τις ψηφιακές υπογραφές RSA. Ωστόσο, μπορούν επίσης να χρησιμοποιηθούν ως το βασικό και μοναδικό δομικό στοιχείο για τη δημιουργία υπογραφών.

Οι υπογραφές βασισμένες σε συνόψεις (Hash-based signatures – HBS) αντιπροσωπεύουν μια υποσχόμενη προσέγγιση στην μετά-κβαντική κρυπτογραφία. Προτάθηκαν για πρώτη φορά από τον Ralph C. Merkle το 1979 [6], λίγο μετά την εισαγωγή της έννοιας της κρυπτογραφίας δημοσίου κλειδιού από τους Diffie και Hellman [7]. Πιο πρόσφατα, έχουν δημοσιευθεί σύγχρονες παραλλαγές του πρωτογενούς συστήματος (π.χ. από τους Bernstein et al. [8]) και έχουν λάβει σημαντική αναγνώριση. Το 2018 η Ομάδα Μηχανικής του Διαδικτύου (IETF) δημοσίευσε το RFC8391 που καθορίζει ένα από αυτά τα συστήματα υπογραφών με σύνοψη [9].

Το πιο σημαντικό πλεονέκτημα της προσέγγισης HBS είναι η ασφάλεια έναντι τόσο κλασικών όσο και μετά-κβαντικών επιθέσεων. Πιο συγκεκριμένα, η ασφάλεια των συστημάτων HBS βασίζεται σε γνωστές απαιτήσεις ασφάλειας που σχετίζονται με τις συναρτήσεις σύνοψης, όπως η ανθεκτικότητα σε συγκρούσεις (collision-resistance) και η αντίσταση πρώτου ορίσματος (pre-image resistance). Αυτό αποτελεί ένα σημαντικό πλεονέκτημα σε σύγκριση με άλλα συστήματα υπογραφής (κλασικά και μετά-κβαντικά) για τους δύο παρακάτω λόγους:

- Τα συστήματα υπογραφής που χρησιμοποιούμε βασίζουν την ασφάλειά τους σε δύο πράγματα: την ασφάλεια μιας συνάρτησης σύνοψης και την ασφάλεια κάποιου υποτιθέμενα δύσκολου υπολογιστικού προβλήματος. Η ασφάλεια μιας συνάρτησης σύνοψης είναι απαραίτητη επειδή τέτοια συστήματα υπογραφής λειτουργούν μόνο σε μηνύματα σταθερού μήκους. Για να υπογράψει ένα μήνυμα αυθαίρετου μήκους, είναι απαραίτητο το μήνυμα αρχικά να μετατραπεί σε ένα μήνυμα σταθερού μήκους, μια διαδικασία που συνήθως εκτελείται από μια συνάρτηση σύνοψης ανθεκτική σε συγκρούσεις, και στη συνέχεια να υπογραφεί αυτή η σύνοψη. Το τελευταίο βήμα (υπογραφή) συνεπάγεται ότι το σύστημα βασίζεται επίσης σε ένα μαθηματικό πρόβλημα όπως η δυσκολία της παραγοντοποίησης ακεραίων (RSA), και η επίλυση του προβλήματος διακριτού λογαρίθμου (ECC). Για τα συστήματα HBS, το σενάριο είναι διαφορετικό καθώς βασίζονται αποκλειστικά στην ασφάλεια των συναρτήσεων σύνοψης (τόσο για τη διαδικασία δημιουργίας σύνοψης όσο και για την υπογραφή). Για αυτόν τον λόγο, το HBS κατατάσσεται ως μια τεχνική υπογραφής που βασίζεται σε ελάχιστες υποθέσεις ασφάλειας.
- Η ασφάλεια των συναρτήσεων σύνοψης είναι ένα από τα πιο μελετημένα θέματα στην κρυπτολογία, προσφέροντας έτσι πολύ περισσότερη εμπιστοσύνη από άλλα, λιγότερο μελετημένα προβλήματα. Όσον αφορά τις κβαντικές επιθέσεις, είναι επαρκώς αποδειγμένο ότι οι κβαντικοί αλγόριθμοι επηρεάζουν μόνο οριακά την ασφάλεια των συναρτήσεων σύνοψης όπως φαίνεται από τον αλγόριθμο του Grover [3]. Αξίζει επίσης να αναφερθεί ότι η επιτυχής επίθεση σε μια συνάρτηση σύνοψης δεν καθιστά ανασφαλές κανένα σύστημα HBS. Αντίθετα, δείχνει απλώς ότι αυτή η συνάρτηση σύνοψης δεν είναι κατάλληλη για το HBS (και πιθανώς δεν είναι κατάλληλη για καμία κρυπτογραφική εφαρμογή).

Ένα άλλο πλεονέκτημα των συστημάτων HBS είναι ότι μπορούν να προσαρμοστούν για να πληρούν διαφορετικές απαιτήσεις απόδοσης. Για παράδειγμα, τα συστήματα HBS έχουν παραμέτρους συστήματος που ελέγχουν τις αντισταθμίσεις μεταξύ του μεγέθους της υπογραφής και του κόστους υπογραφής / επαλήθευσης, για το ίδιο επίπεδο ασφάλειας. Αυτό καθιστά το HBS μια ευέλικτη λύση που μπορεί να επιτύχει ταχύτερη επεξεργασία με το κόστος μεγαλύτερων υπογραφών, και το αντίστροφο καθώς τα συστήματα HBS μπορούν να

ρυθμιστούν ώστε να μειώσουν τον χρόνο που απαιτείται για τη δημιουργία ή την επαλήθευση μιας υπογραφής (δηλ. να βελτιώσουν την ταχύτητα της επεξεργασίας). Ωστόσο, αυτή η αύξηση της ταχύτητας έρχεται με το συμβιβασμό ότι οι υπογραφές που παράγονται θα είναι μεγαλύτερες σε μέγεθος (δηλ. θα καταλαμβάνουν περισσότερο χώρο). Επιπλέον, η επιλογή της συνάρτησης σύνοψης που χρησιμοποιείται στο σύστημα μπορεί να επηρεάσει τόσο την ταχύτητα λειτουργίας όσο και το επίπεδο ασφάλειας, προσφέροντας έτσι τη δυνατότητα προσαρμογής του συστήματος ανάλογα με τις ανάγκες.

Από την άλλη πλευρά, τα πιο αποδοτικά συστήματα HBS είναι τα συστήματα HBS με διατήρηση κατάστασης (stateful HBS schemes), δηλαδή συστήματα με διατήρηση πληροφοριών κατάστασης. Σε αυτά τα συστήματα ο υπογράφων πρέπει να αλλάζει το ιδιωτικό κλειδί μετά από κάθε υπογραφή, αλλιώς το κρυπτοσύστημα δε θεωρείται ασφαλές. Αυτό σημαίνει ότι πρέπει να υπάρχουν διαδικασίες διαχείρισης των πληροφοριών «κατάστασης» για να αποτραπεί η επαναχρησιμοποίηση των ιδιωτικών κλειδιών. Ανάλογα με την εφαρμογή, η υλοποίηση μιας διαδικασίας διαχείρισης κατάστασης μπορεί να αποτελεί πρόβλημα. Για τα συστήματα με διατήρηση κατάστασης, το NIST έχει δημοσιεύσει το SP 800-208 [10] που τυποποιεί το LMS [11] και το XMSS [12, 9], δύο συστήματα υπογραφών με διατήρηση κατάστασης που βασίζονται σε συναρτήσεις σύνοψης.

Τα συστήματα HBS χωρίς διατήρηση κατάστασης (stateless HBS schemes) [13], δηλ. συστήματα χωρίς διατήρηση πληροφοριών κατάστασης, λειτουργούν ως κανονικές υπογραφές καθώς δεν χρειάζονται αλλαγή των ιδιωτικών κλειδιών, αποφεύγοντας έτσι το σχετικό διαχειριστικό κόστος. Ωστόσο, αυτό συνοδεύεται από το κόστος μεγαλύτερων υπογραφών και μεγαλύτερων χρόνων επεξεργασίας σε σχέση με τα συστήματα με διατήρηση κατάστασης. Παραδείγματα συστημάτων χωρίς διατήρηση κατάστασης αποτελούν τα eXtended Merkle Signature Scheme (XMSS), Forest of Random Subsets (FORS), και SPHINCS+, μια παραλλαγή του οποίου ορίζεται στο FIPS 205 [13] ως Stateless Hashed-based Digital Signature Algorithm (SLH-DSA).

15.3.1.1 Υπογραφές μιας Χρήσης

Τα συστήματα υπογραφών μιας χρήσης (one-time signatures - ΟΤΣ) αποτελούν σημαντικά δομικά στοιχεία για την κατασκευή πρακτικών συστημάτων υπογραφών πολλαπλών χρήσεων που βασίζονται στις συνόψεις. Συνήθως βρίσκουν περιορισμένη εφαρμογή λόγω του κύριου περιορισμού τους: ένα κλειδί υπογραφής δεν πρέπει ποτέ να χρησιμοποιηθεί για την υπογραφή περισσότερων του ενός μηνυμάτων. Αυτό συμβαίνει επειδή δύο διαφορετικές υπογραφές που δημιουργούνται με το ίδιο ιδιωτικό κλειδί παρέχουν αρκετές πληροφορίες σε έναν επιτιθέμενο για να δημιουργήσει πλαστές υπογραφές. Λόγω αυτού του περιορισμού δε χρησιμοποιούνται ως αυτόνομη λύση αλλά ως μέρος ενός μεγαλύτερου συστήματος, όπως σε συστήματα πολλαπλών χρήσεων που βασίζονται στις συνόψεις.

Τα συστήματα που περιγράφονται σε αυτή την ενότητα είναι τα συστήματα Lamport, Winternitz και WOTS+. Το σύστημα Lamport δεν έχει πρακτική εφαρμογή, αλλά περιγράφεται για να διευκολύνει την κατανόηση της έννοιας των υπογραφών μιας χρήσης και των δύο τελευταίων συστημάτων.

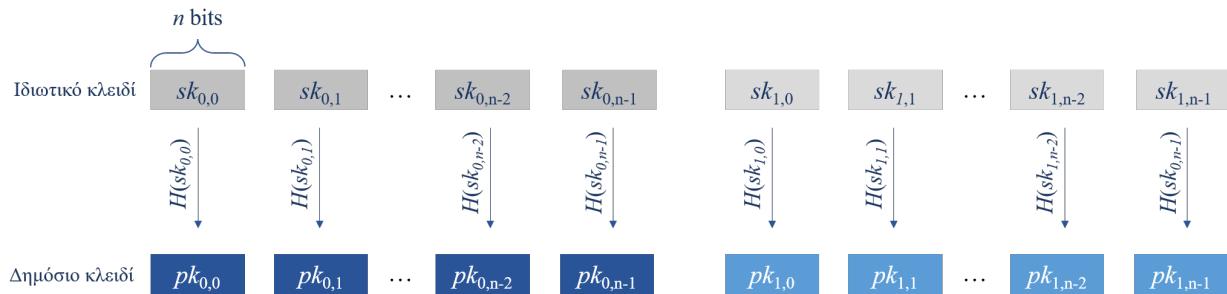
Σύστημα Lamport: Πρόκειται για ένα σύστημα υπογραφής μιας χρήσης που βασίζεται αποκλειστικά σε συναρτήσεις σύνοψης [14]. Σε γενικές γραμμές, χρησιμοποιεί δύο ακολουθίες τυχαίων τιμών ως το ιδιωτικό κλειδί και δύο ακολουθίες που αποτελούνται από τις κατακερματισμένες τιμές αυτών των τυχαίων τιμών ως το δημόσιο κλειδί. Η υπογραφή ενός μηνύματος αποτελείται από τη μηνύματα του ιδιωτικού κλειδιού. Αν κάποιο bit του μηνύματος είναι 0, η υπογραφή περιλαμβάνει ένα συγκεκριμένο μέρος του ιδιωτικού κλειδιού από την πρώτη ακολουθία τυχαίων τιμών. Αν το bit είναι 1, τότε χρησιμοποιείται ένα τμήμα του ιδιωτικού κλειδιού από τη δεύτερη ακολουθία. Επομένως, η υπογραφή αποτελείται από διάφορα τμήματα του ιδιωτικού κλειδιού που προέρχονται είτε από την πρώτη ακολουθία είτε από τη δεύτερη ακολουθία, ανάλογα με τα bits του μηνύματος προς υπογραφή. Η διαδικασία επαλήθευσης περιλαμβάνει τον υπολογισμό της σύνοψης των τμημάτων του ιδιωτικού κλειδιού που χρησιμοποιήθηκαν στην υπογραφή και τη σύγκρισή τους με τις αντίστοιχες τιμές του δημόσιου κλειδιού, ανάλογα με τα bits του μηνύματος. Δεδομένου ότι η λειτουργία σύνοψης είναι μη αναστρέψιμη, ένα τέτοιο σύστημα είναι ασφαλές. Ταυτόχρονα όμως, ένα ζεύγος κλειδιών δεν πρέπει ποτέ να

χρησιμοποιηθεί δύο φορές, καθώς η υπογραφή αποκαλύπτει πραγματικά τμήματα του ιδιωτικού κλειδιού.

Σε μια πιο τυπική περιγραφή, το ιδιωτικό κλειδί στο σύστημα Lamport αποτελείται από δύο ακολουθίες δυαδικών τιμών, η καθεμία από τις οποίες έχει μήκος n^2 bits: $(sk_0, sk_1) \in \{0, 1\}^{2n^2}$. Αυτό σημαίνει ότι το ιδιωτικό κλειδί περιλαμβάνει δύο ξεχωριστές ακολουθίες τυχαίων bits, με συνολικό μήκος $2n^2$, δηλαδή n^2 bits για κάθε ακολουθία, που επιλέγονται τυχαία. Το δημόσιο κλειδί αποτελείται επίσης από δύο ακολουθίες των n^2 bits η κάθε μία: $(pk_0, pk_1) \in \{0, 1\}^{2n^2}$, οι οποίες υπολογίζονται ως:

$$pk_{(i,j)} \leftarrow H(sk_{(i,j)}), \quad \text{για } 0 \leq i < 2, 0 \leq j < n$$

Η παραπάνω εξίσωση σημαίνει ότι για κάθε θέση (i, j) του δημόσιου κλειδιού, υπολογίζεται η σύνοψη της αντίστοιχης θέσης του ιδιωτικού κλειδιού με τη χρήση της συνάρτησης σύνοψης H . Το Σχήμα 15.1 απεικονίζει αυτή τη διαδικασία δημιουργίας κλειδιού.



Σχήμα 15.1: Δημιουργία κλειδιών στο σύστημα Lamport.

Η διαδικασία δημιουργίας υπογραφής ενός μηνύματος M' οποιουδήποτε μήκους ξεκινά με τον υπολογισμό της σύνοψης του μηνύματος: $M \leftarrow H(M')$. Η υπογραφή υπολογίζεται ως: $S_i \leftarrow sk(M_{i,i})$, για $0 \leq i < n$. Αυτό σημαίνει ότι, για κάθε bit i της σύνοψης του μηνύματος M , το i -οστό τμήμα της υπογραφής S_i καθορίζεται από την τιμή του i -οστού bit της σύνοψης. Αν το bit είναι 0, τότε η υπογραφή παίρνει το αντίστοιχο τμήμα από την ακολουθία sk_0 του ιδιωτικού κλειδιού, ενώ αν είναι 1, η υπογραφή παίρνει το αντίστοιχο τμήμα από την ακολουθία sk_1 .

Για παράδειγμα, αν το i -οστό bit του M είναι: $M_i = 1$, τότε $S_i = sk_{1,i}$. Όπως βλέπουμε, η υπογραφή αποτελείται από τμήματα του ιδιωτικού κλειδιού (τα bits της σύνοψης του μηνύματος καθορίζουν αν τα τμήματα προέρχονται από το sk_0 ή το sk_1). Η διαδικασία επαλήθευσης της υπογραφής συνίσταται στον υπολογισμό της σύνοψης του μηνύματος $M \leftarrow H(M')$, και στη συνέχεια γίνεται δεκτή αν: $pk_i == H(S_i)$, για όλα τα $0 \leq i < n$. Δηλαδή, το i -οστό τμήμα του δημόσιου κλειδιού (pk_i) πρέπει να είναι ίσο με την σύνοψη $H(S_i)$ του αντίστοιχου τμήματος της υπογραφής (S_i). Με άλλα λόγια, η επαλήθευση λειτουργεί υπολογίζοντας τη σύνοψη κάθε τμήματος της υπογραφής και συγκρίνοντάς την με την αντίστοιχη τιμή του δημόσιου κλειδιού. Αν όλοι οι συνόψεις αντιστοιχούν σωστά, τότε η υπογραφή θεωρείται έγκυρη. Το Σχήμα 15.2 απεικονίζει τη διαδικασία υπογραφής Lamport για μια υποθετική σύνοψη μηνύματος M μήκους n bits.



Σχήμα 15.2: Δημιουργία υπογραφής στη σύνοψη M με το σύστημα Lamport. ;

Οι διαδικασίες δημιουργίας υπογραφής και επαλήθευσης στο σύστημα Lamport είναι πολύ γρήγορες καθώς απαιτούν μόνο μια σύνοψη ανά bit της σύνοψης του μηνύματος. Από την άλλη πλευρά, το σύστημα Lam-

port θεωρείται μη πρακτικό λόγω των μεγάλων μεγεθών του δημοσίου κλειδιού και της υπογραφής που αυξάνονται τετραγωνικά με το n .

Σύστημα Winternitz: Αποτελεί ένα σημαντικά πιο πρακτικό σύστημα από αυτό του Lamport [6]. Σε γενικές γραμμές, η σύνοψη του μηνύματος αντιμετωπίζεται ως μια ακολουθία ακέραιων και κάθε ακέραιος προσδιορίζει πόσες φορές πρέπει να εφαρμοστεί η συνάρτηση σύνοψης σε ένα τμήμα του ιδιωτικού κλειδιού για να παραχθεί ένα τμήμα της υπογραφής. Για παράδειγμα, ας υποθέσουμε ότι ο πρώτος ακέραιος στην αναπαράσταση του μηνύματος είναι k , όπου $0 \leq k \leq N$, και το N είναι ο μέγιστος αριθμός που μπορεί να λάβει το k . Το πρώτο τμήμα της υπογραφής υπολογίζεται ως το αποτέλεσμα K διαδοχικών κλήσεων της συνάρτησης στη συνάρτηση σύνοψης (με τον όρο διαδοχική εννοούμε ότι η έξοδος μιας κλήσης χρησιμοποιείται ως είσοδος στην επόμενη κλήση) στο αντίστοιχο τμήμα του ιδιωτικού κλειδιού. Το δημόσιο κλειδί είναι το αποτέλεσμα της εφαρμογής της συνάρτησης σύνοψης N επαναλαμβανόμενες φορές σε κάθε τμήμα του ιδιωτικού κλειδιού. Επομένως, για να επαληθεύσει κανείς αν ένα τμήμα της υπογραφής είναι έγκυρο, αρκεί να υπολογίσει $(N - k)$ κλήσεις στη συνάρτηση σύνοψης και να ελέγξει αν το αποτέλεσμα ταιριάζει με το αντίστοιχο τμήμα του δημόσιου κλειδιού.

Αναλυτικότερα, δεδομένων των παραμέτρων w, L, l_1 και l_2 , το ιδιωτικό κλειδί αποτελείται από μια ακολουθία L τυχαία επιλεγμένων τμημάτων των n bits το καθένα: $sk \in \{0,1\}^{Ln}$. Για να υπολογίσουμε το δημόσιο κλειδί, αρχικά υπολογίζουμε:

$$pk_i \leftarrow H_n(sk_i)^{2^w-1}, \quad \text{για } 0 \leq i < L$$

Αυτό σημαίνει ότι το i -οστό τμήμα του δημόσιου κλειδιού pk_i προκύπτει από την επαναλαμβανόμενη εφαρμογή της συνάρτησης σύνοψης H_n στο αντίστοιχο τμήμα του ιδιωτικού κλειδιού sk_i ; $2^w - 1$ φορές. Στη συνέχεια, το δημόσιο κλειδί υπολογίζεται ως:

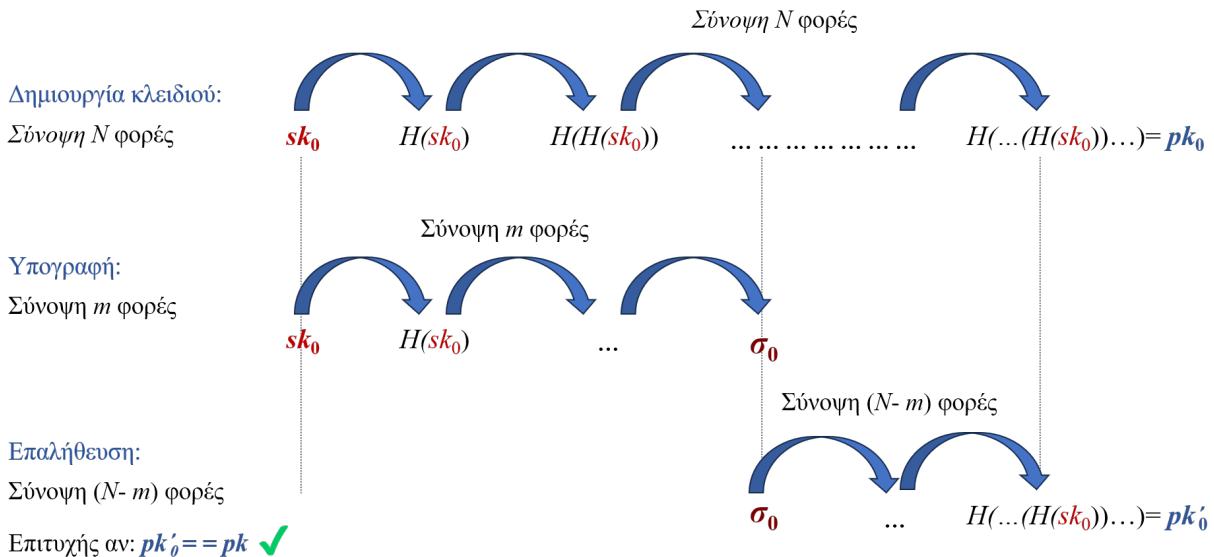
$$pk \leftarrow H_n(pk_1 \| \dots \| pk_L)$$

Η υπογραφή ενός μηνύματος αυθαίρετου μήκους M'' απαιτεί μερικά ενδιάμεσα βήματα για να παραχθεί η αναπαράσταση του μηνύματος. Αρχικά, υπολογίζεται μια προκαταρκτική σύνοψη του μηνύματος: $M' \leftarrow H_m(M'')$. Η τιμή M' αντιμετωπίζεται ως l_1 μπλοκ των w bits το καθένα, δηλαδή l_1 ακέραιοι στο διάστημα $[0..2^w - 1]$. Στη συνέχεια, υπολογίζεται ένα άθροισμα ελέγχου CS του M' : $CS \leftarrow \sum(2^w - 1 - M'_i)$, για $0 \leq i < l_1$. Το άθροισμα ελέγχου είναι σημαντικό για να αποφευχθούν ορισμένες επιθέσεις παραποίησης. Το μέγεθος του CS σε μπλοκ των w bits το καθένα είναι: $l_2 \leftarrow \lceil \log(l_1(2^w - 1))/w \rceil$. Η παράμετρος L ορίζεται ως: $L \leftarrow l_1 + l_2$. Η αναπαράσταση του μηνύματος M'' υπολογίζεται τελικά ως η συνένωση του M' και του αθροίσματος ελέγχου του: $M \leftarrow (M' \| CS)$.

Η υπογραφή υπολογίζεται ως: $S = [S_0, \dots, S_L]$, όπου $S_i \leftarrow H_n(sk_i)^{2^w-1-M_i}$, για $0 \leq i < L$. Η διαδικασία επαλήθευσης είναι πολύ παρόμοια με την υπογραφή και ξεκινά με τον υπολογισμό του M' από το M'' και στη συνέχεια την αναπαράσταση μηνύματος M , όπως περιγράφεται παραπάνω. Η υπογραφή είναι έγκυρη αν: $pk = H_n(t_1 \| \dots \| t_L)$, όπου $t_i \leftarrow H_n(S_i)^{M_i}$. Το Σχήμα 15.3 δείχνει τη διαδικασία δημιουργίας κλειδιών, υπογραφής και επαλήθευσης αυτών για το σύστημα Winternitz για το τμήμα 0 (σχετικό με sk_0, pk_0 και S_0), για μια αναπαράσταση μηνύματος m . Αυτό οδηγεί σε ένα σύστημα που είναι πολύ πιο αποδοτικό από το σύστημα Lamport, δεδομένου ότι το μέγεθος του δημόσιου κλειδιού αυξάνεται μόνο γραμμικά (αντί για τετραγωνικά).

Σύστημα WOTS+ (Winternitz One-Time Signature Plus): Αποτελεί μια βελτίωση του συστήματος Winternitz, καθώς απαιτεί πιο ήπιες απαιτήσεις ασφάλειας όπως η αντίσταση δεύτερου ορίσματος (second pre-image resistance) αντί της ανθεκτικότητας σε συγκρούσεις [15, 13]. Αυτό επιτρέπει τη χρήση συναρτήσεων σύνοψης με μικρότερο μήκος σύνοψης (δηλ. μικρότερη παράμετρο n) για το ίδιο επίπεδο ασφάλειας, οδηγώντας σε μικρότερες υπογραφές.

Το WOTS+ βασίζεται στην αντίσταση δεύτερου ορίσματος επειδή χρησιμοποιεί μια τεχνική κλήσεων σύνοψης με τυχαίο συντελεστή που μειώνει τις πιθανότητες συγκρούσεων. Στην πράξη, η είσοδος σε κάθε κλήση σύνοψης περνάει πρώτα από μια συνάρτηση αποκλειστικού-Η (XOR) με μερικά τυχαία bits που ονομάζονται



Σχήμα 15.3: Δημιουργία κλειδιών και υπογραφής και επαλήθευση υπογραφής με το σύστημα Winternitz.

bitmasks. Αυτή η λειτουργία περιλαμβάνεται στον ορισμό της συνάρτησης αλυσίδας WOTS+, η οποία καλείται από τις λειτουργίες δημιουργίας κλειδιών, υπογραφής και επαλήθευσης του WOTS+ αντί για άμεσες κλήσεις στη συνάρτηση σύνοψης (όπως γίνεται στα συστήματα Lamport και Winternitz). Η συνάρτηση αλυσίδας (chaining function) WOTS+, c , περιγράφεται τυπικά ως εξής:

$$\begin{aligned} c_k^i(x, r) &\leftarrow x, \text{ αν } i = 0 \\ c_k^i(x, r) &\leftarrow f_k(c_k^{i-1}(x, r) \oplus r_i), \text{ αν } i > 0 \end{aligned}$$

Η πρώτη εξίσωση ορίζει ότι αν $i = 0$, η συνάρτηση αλυσίδας απλά επιστρέφει την είσοδο x χωρίς περαιτέρω μετασχηματισμούς. Αυτό αποτελεί την αρχική τιμή της συνάρτησης αλυσίδας. Η δεύτερη εξίσωση ορίζει ότι για $i > 0$, η συνάρτηση αλυσίδας εφαρμόζει την κρυπτογραφική συνάρτηση f_k στην τιμή που προέκυψε από τον προηγούμενο γύρο της αλυσίδας, αφού πρώτα εκτελεστεί το bitwise XOR της τιμής αυτής με το τυχαίο bitmask r_i . Αυτή η επαναλαμβανόμενη διαδικασία εξασφαλίζει μεγαλύτερη ασφάλεια τυχαιοποιώντας την είσοδο πριν την εφαρμογή της συνάρτησης f_k . Η f_k είναι συνάρτηση που ανήκει σε μια οικογένεια συναρτήσεων $F_k = \{f_k : \{0,1\}^n \rightarrow \{0,1\}^m | k \in K\}$, με το χώρο κλειδιών K_n , και r αντιπροσωπεύει το σύνολο των bitmasks $r = (r_1, \dots, r_j) \in \{0,1\}^{n \times j}$ που χρειάζονται για την τυχαιοποίηση των κλήσεων της συνάρτηση σύνοψης. Στην πράξη, η f_k μπορεί να θεωρηθεί ως μια συνάρτηση σύνοψης με κλειδί χωρίς συμπίεση, που απεικονίζει n bits εισόδου σε n bits εξόδου. Στον γύρο i , η συνάρτηση αλυσίδας αξιολογεί την f_k στην τιμή του XOR των bitwise της $(i-1)$ -ης κλήσης αλυσίδας με το τυχαίο bitmask r_i .

Η διαδικασία δημιουργίας κλειδιών στο WOTS+ είναι παρόμοια με εκείνη του συστήματος Winternitz, αλλά εδώ χρησιμοποιείται η συνάρτηση αλυσίδας $c_k^i(x, r)$ αντί για άμεση κλήση στη συνάρτηση σύνοψης. Η διαδικασία ξεκινά με την τυχαία επιλογή $(L + 2^w - 1)$ ακολουθών n bits.

Από αυτές τις ακολουθίες, οι L πρώτες θα χρησιμοποιηθούν ως το ιδιωτικό κλειδί (ή κλειδί υπογραφής) και αναπαρίστανται ως:

$$sk = (sk_1, sk_2, \dots, sk_L)$$

Οι υπόλοιπες $2^w - 1$ ακολουθίες θα χρησιμοποιηθούν για τη δημιουργία των λεγόμενων *bitmasks*, τα οποία προσφέρουν επιπλέον ασφάλεια κατά τη διαδικασία.

Το δημόσιο κλειδί υπολογίζεται εφαρμόζοντας τη συνάρτηση αλυσίδας $c_{i,k}(x, r)$ σε κάθε κομμάτι του ιδιωτικού κλειδιού για $2^w - 1$ επαναλήψεις. Συγκεκριμένα, για κάθε τμήμα του ιδιωτικού κλειδιού, το δημόσιο

κλειδί υπολογίζεται ως:

$$\text{pk}_i \leftarrow c_{2^w-1,k}(\text{sk}_i, \text{r})$$

Όταν ολοκληρωθεί η διαδικασία για όλα τα τμήματα του ιδιωτικού κλειδιού, το συνολικό δημόσιο κλειδί υπολογίζεται ως:

$$\text{pk} \leftarrow H(\text{pk}_1 \parallel \text{pk}_2 \parallel \dots \parallel \text{pk}_L)$$

όπου H είναι η συνάρτηση σύνοψης που συμπυκνώνει όλα τα επιμέρους δημόσια κλειδιά σε ένα.

Για την υπογραφή ενός μηνύματος M'' , αρχικά το μήνυμα πρέπει να μετατραπεί σε μια κατάλληλη μορφή, που ονομάζεται αναπαράσταση μήκους M . Αυτή η αναπαράσταση αποτελείται από L τμήματα.

Η υπογραφή υπολογίζεται εφαρμόζοντας τη συνάρτηση αλυσίδας σε κάθε τμήμα του ιδιωτικού κλειδιού. Συγκεκριμένα, για κάθε τμήμα t_i της αναπαράστασης του μηνύματος M_i , η υπογραφή υπολογίζεται ως:

$$S_i \leftarrow c_{2^w-1-M_i,k}(\text{sk}_i, \text{r})$$

όπου $1 \leq i \leq L$. Αυτό σημαίνει ότι για κάθε κομμάτι t_i της υπογραφής, η συνάρτηση αλυσίδας εφαρμόζεται διαφορετικό αριθμό φορών, ανάλογα με την τιμή του αντίστοιχου τμήματος t_i της αναπαράστασης του μηνύματος.

Για την επαλήθευση της υπογραφής, ξεκινάμε υπολογίζοντας την αναπαράσταση μήκους M του μηνύματος M'' . Στη συνέχεια, η υπογραφή είναι έγκυρη αν το συνολικό δημόσιο κλειδί που προκύπτει από τα τμήματα t_i είναι ίσο με το αρχικό δημόσιο κλειδί. Συγκεκριμένα, για κάθε t_i , υπολογίζεται ως εξής:

$$t_i \leftarrow c_{M_i,k}(S_i, \text{r})$$

Η υπογραφή θεωρείται έγκυρη αν:

$$\text{pk} = H(t_1 \parallel t_2 \parallel \dots \parallel t_L)$$

Δηλαδή, η σύνοψη όλων των τιμών t_i πρέπει να ταιριάζει με το δημόσιο κλειδί που είχε υπολογιστεί κατά τη διαδικασία δημιουργίας του κλειδιού.

Το WOTS+ προσφέρει βελτιωμένη ασφάλεια και αποδοτικότητα σε σχέση με το αρχικό σύστημα Winter-nitz, κάνοντάς το πιο πρακτικό για χρήση σε σύγχρονα συστήματα υπογραφής. Τα κλειδιά και οι υπογραφές στο WOTS+ έχουν πιο μικρό μήκος, ενώ η διαδικασία επαλήθευσης παραμένει απλή και αποτελεσματική.

15.3.1.2 Από Υπογραφές μιας Χρήσης σε Υπογραφές Πολλαπλών Χρήσεων

Όπως αναφέρθηκε προηγουμένως, οι υπογραφές μιας χρήσης έχουν περιορισμένη εφαρμογή λόγω του γεγονότος ότι το ιδιωτικό κλειδί δεν πρέπει να χρησιμοποιηθεί για την υπογραφή περισσότερων από ένα μηνυμάτων. Αυτό περιορίζει την ευρεία υιοθέτηση αυτών των συστημάτων παρά την απλότητά τους και τις ισχυρές εγγυήσεις ασφαλείας που προσφέρουν.

Ωστόσο, υπάρχουν μερικοί τρόποι για να δημιουργήσουμε συστήματα υπογραφών πολλαπλών χρήσεων από συστήματα υπογραφών μιας χρήσης. Η απλή προσέγγιση θα ήταν να χρησιμοποιήσουμε πολλαπλά ζεύγη κλειδιών μιας χρήσης και να χρησιμοποιήσουμε κάθε ιδιωτικό κλειδί για να υπογράψουμε μόνο ένα μήνυμα. Μόλις χρησιμοποιηθεί ένα ιδιωτικό κλειδί, ο υπογράφων πρέπει να το απορρίψει για να αποτρέψει την επαναχρησιμοποίησή του. Αυτή η προσέγγιση θα λειτουργούσε αλλά με ένα σημαντικό μειονέκτημα: η οντότητα που κάνει την επαλήθευση θα πρέπει είτε να αποθηκεύσει όλα τα δημόσια κλειδιά μιας χρήσης (μια μη αποδοτική από άποψη μνήμης προσέγγιση) είτε να έχει ένα μηχανισμό για τη συνεχή παροχή νέων δημόσιων κλειδιών μιας χρήσης όπως απαιτείται (μια προσέγγιση που σπάνια είναι βιώσιμη). Συνοπτικά, αυτή η προσέγγιση δεν είναι επεκτάσιμη.

Η λύση που πρότεινε ο Ralph C. Merkle [6] είναι επεκτάσιμη καθώς συνδέει πολλά δημόσια κλειδιά μιας χρήσης με ένα μόνο δημόσιο κλειδί πολλαπλών χρήσεων. Η ιδέα συνίσταται στην κατασκευή ενός δυαδικού δέντρου, όπως αναλύθηκε στην Ενότητα 3.2.6 το οποίο έχει τα δημόσια κλειδιά μιας χρήσης ως «φύλλα». Οι υπόλοιποι κόμβοι του δέντρου κατασκευάζονται ως οι κατακερματισμένες τιμές της συνένωσης των δύο

θυγατρικών κόμβων. Εφαρμόζοντας αυτόν τον κανόνα μέχρι την κορυφή οδηγεί σε έναν κόμβο-ρίζα που χρησιμοποιείται ως το δημόσιο κλειδί πολλαπλών χρήσεων. Αυτή είναι η βασική ιδέα που παρουσιάζεται στο Σύστημα Υπογραφής Merkle (γνωστό και ως σύστημα LDWM). Όλα τα άλλα σημαντικά συστήματα HBS πολλαπλών χρήσεων αξιοποιούν αυτή την τεχνική με κάποιες μικρές διαφορές.

Συνοπτικά, υπάρχουν αποδοτικοί τρόποι για να κατασκευαστούν υπογραφές πολλαπλών χρήσεων με βάση τις υπογραφές μιας χρήσης, οι οποίες με τη σειρά τους βασίζονται στις συναρτήσεις σύνοψης. Στο Σχήμα 15.4 απεικονίζεται μια μη εξαντλητική λίστα πιθανών επιλογών αναφορικά με τις ψηφιακές υπογραφές βασισμένες σε συνόψεις.



Σχήμα 15.4: Σύνθεση των συστημάτων υπογραφών που βασίζονται σε συνόψεις.

Αξίζει να σημειωθεί πως πέραν των υπογραφών μιας χρήσης και αυτών των πολλαπλών χρήσεων, υπάρχουν και οι υπογραφές μερικών χρήσεων (few-times signatures). Σε αντίθεση με τις υπογραφές μιας χρήσης, ένα σχήμα υπογραφών μερικών χρήσεων μπορεί να χρησιμοποιηθεί για την υπογραφή μηνυμάτων για μερικές φορές μόνο, καθώς κάθε φορά που χρησιμοποιείται, ορισμένες πληροφορίες εκτίθενται, μειώνοντας την ασφάλεια του κλειδιού. Τέτοιο σχήμα είναι το HORST (Hash to Obtain Random Subset with Trees) [16].

15.3.1.3 SPHINCS

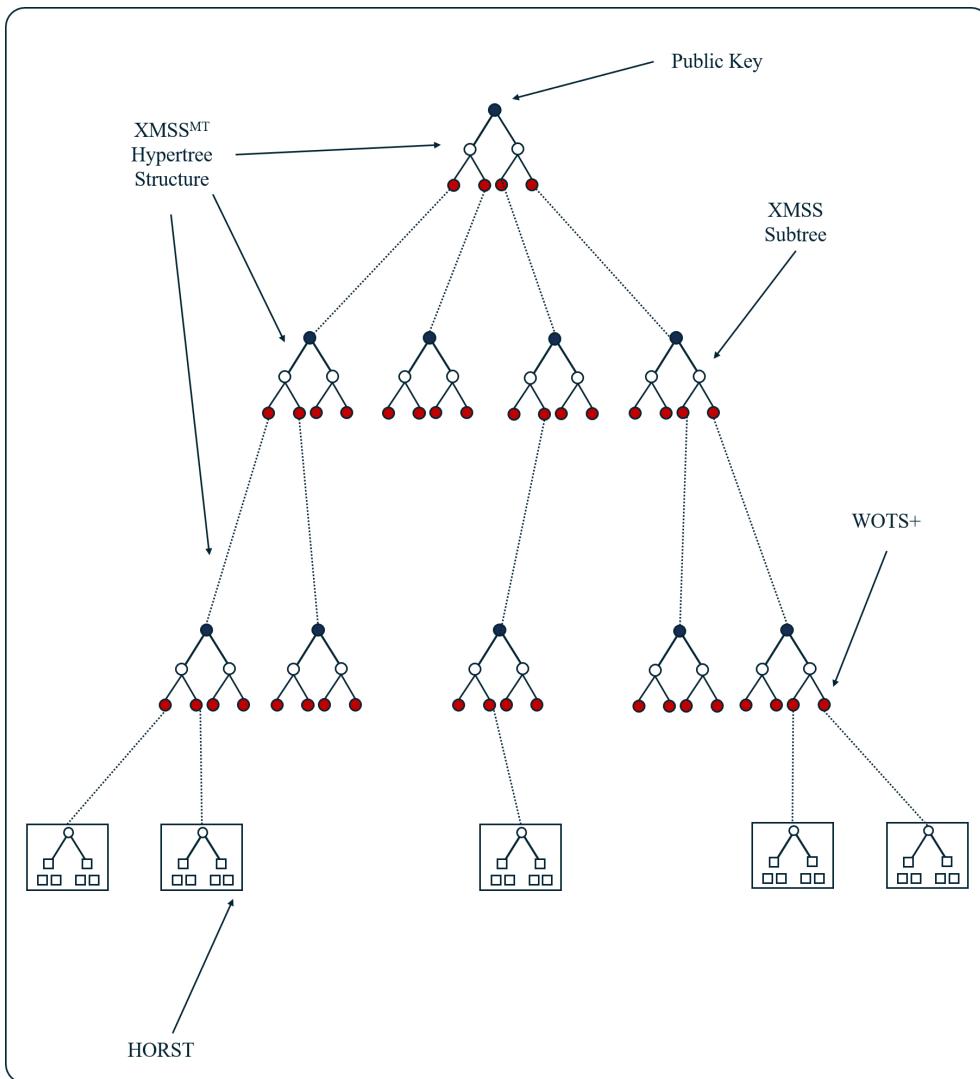
Το σύστημα SPHINCS [8] αποτελείται από τα δομικά στοιχεία WOTS+ [15], XMSS [9] και HORST [16]. Σε αντίθεση με άλλα συστήματα όπως τα WOTS+ και HORST που για να χρησιμοποιήσουν έναν μεγάλο αριθμό κλειδιών είναι απαραίτητο να δημιουργηθούν τα κλειδιά εκ των προτέρων για να υπολογιστεί η ρίζα του δέντρου Merkle, το SPHINCS μπορεί να διαχειριστεί έναν πολύ μεγαλύτερο αριθμό κλειδιών χωρίς την ανάγκη προϋπολογισμού όλων των φύλλων, χρησιμοποιώντας δύο μεθόδους:

- Υπερ-δέντρο (Hyper-Tree)
- Σύστημα τυχαίας διευθυνσιοδότησης μονοπατιού κλειδιών (Random Key Path Addressing Scheme)

Μια γενική σχηματική αναπαράσταση της δομής του δέντρου SPHINCS απεικονίζεται στο Σχήμα 15.5, ενώ το Σχήμα 15.6 απεικονίζει την ίδια δομή με περισσότερες λεπτομέρειες.

Ο Πίνακας 15.1 αποτυπώνει το σύνολο των συναρτήσεων που χρησιμοποιούνται από το σύστημα SPHINCS.

Η δομή υπερ-δέντρου στο SPHINCS είναι ένα σύστημα πολλαπλών δέντρων τοποθετημένων το ένα πάνω στο άλλο, σχηματίζοντας ένα πολυεπίπεδο δέντρο. Το συνολικό ύψος αυτού του υπερ-δέντρου συμβολίζεται ως h , και το υπερ-δέντρο χωρίζεται σε διάφορα επίπεδα. Κάθε επίπεδο περιλαμβάνει ένα ξεχωριστό δέντρο με συγκεκριμένο ύψος. Ο συνολικός αριθμός επιπέδων του υπερ-δέντρου είναι d , και κάθε επιμέρους δέντρο έχει ύψος h/d , που σημαίνει ότι όλα τα επιμέρους δέντρα έχουν το ίδιο ύψος και συμβάλλουν στο συνολικό ύψος του υπερ-δέντρου.

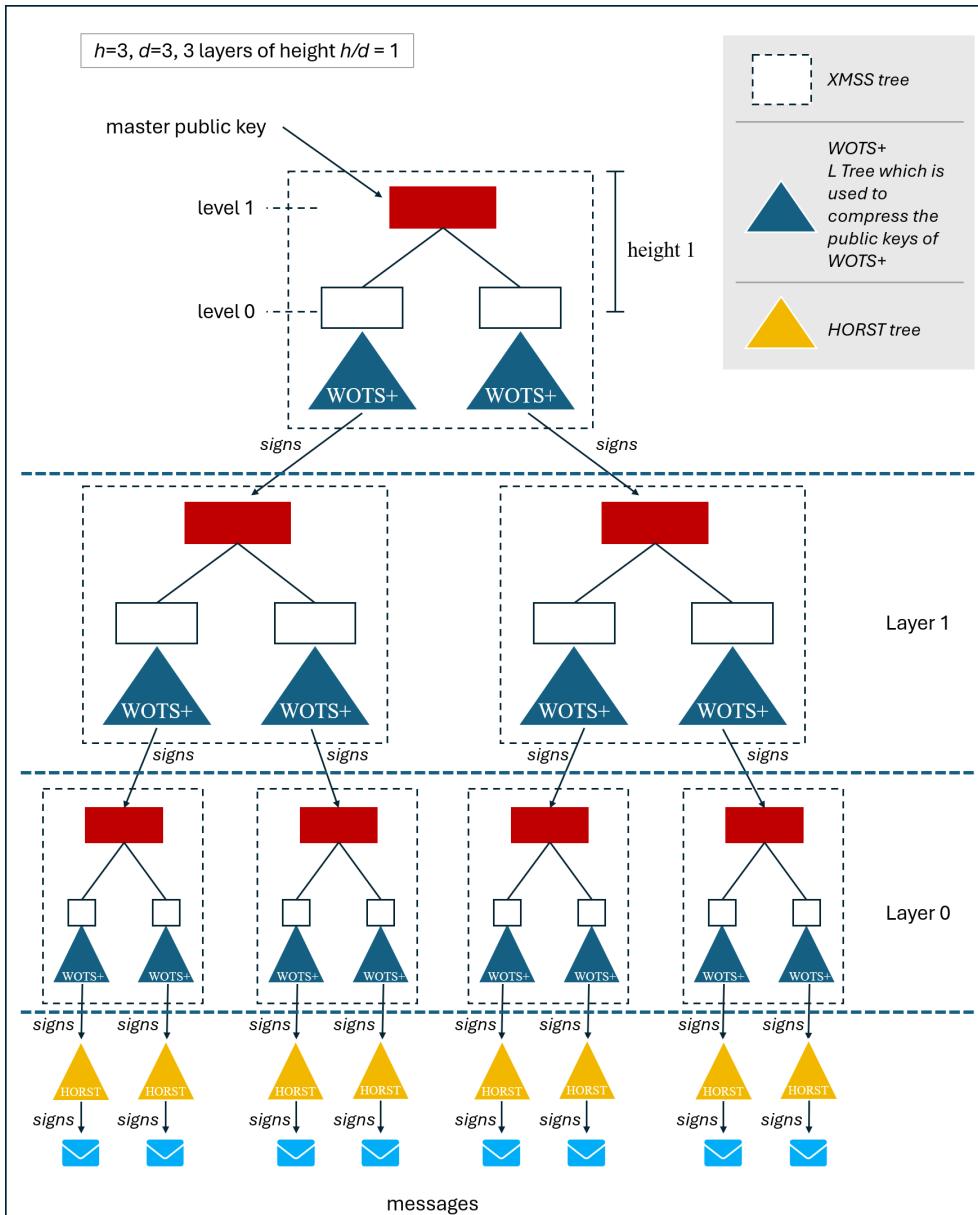


Σχήμα 15.5: Δομή δέντρου του SPHINCS (γενική αναπαράσταση).

Πίνακας 15.1: Συναρτήσεις που χρησιμοποιούνται στο SPHINCS.

Συνάρτηση	Περιγραφή
$F : \{0,1\}^n \rightarrow \{0,1\}^n$	Συνάρτηση σύνοψης (χωρίς συμπίεση)
$H : \{0,1\}^{2n} \rightarrow \{0,1\}^n$	Συνάρτηση σύνοψης (με συμπίεση)
$H : \{0,1\}^n \times \{0,1\}^* \rightarrow \{0,1\}^m$	Συνάρτηση σύνοψης (με συμπίεση) αυθαίρετης εισόδου
$G_\lambda : \{0,1\}^n \rightarrow \{0,1\}^{\lambda n}$	Οικογένεια ψευδοτυχαίων γεννητριών
$F_\lambda : \{0,1\}^\lambda \{0,1\}^n \rightarrow \{0,1\}^n$	Σύνολο οικογενειών ψευδοτυχαίων συναρτήσεων
$F : \{0,1\}^* \{0,1\}^n \rightarrow \{0,1\}^{2n}$	Συνάρτηση σύνοψης (με συμπίεση) με αυθαίρετον μήκους είσοδο

Το υπερ-δέντρο αυτό δε δημιουργείται εξ ολοκλήρου από την αρχή. Αντίθετα, τα επιμέρους δέντρα δημιουργούνται δυναμικά καθώς απαιτείται η υπογραφή ή η επαλήθευση ενός μηνύματος, επιτρέποντας έτσι τη διαχείριση πολύ μεγάλου αριθμού κλειδιών χωρίς τον υπολογισμό όλων των δέντρων εκ των προτέρων. Συγκεκριμένα, όταν υπογράφεται ένα μήνυμα χρειάζεται να δημιουργηθούν μόνο τα δέντρα κατά μήκος μιας συγκεκριμένης διαδρομής στο υπερ-δέντρο. Στο χαμηλότερο επίπεδο του υπερ-δέντρου SPHINCS, υπάρχει ένα επίπεδο δέντρων HORST που περιέχουν ιδιωτικά κλειδιά που χρησιμοποιούνται για την υπογραφή μηνυμάτων. Όταν ένα μήνυμα πρέπει να υπογραφεί, το SPHINCS επιλέγει ένα δέντρο HORST για να υπογράψει



Σχήμα 15.6: Δομή δέντρου του SPHINCS (σχηματική αναπαράσταση).

το μήνυμα και δημιουργεί μια υπογραφή S_H . Πάνω από το επίπεδο HORST, το οποίο είναι το επίπεδο 0, υπάρχουν L -δέντρα που αποτελούνται από ζεύγη κλειδιών WOTS+. Κάθε φύλλο αυτών των δέντρων περιέχει τις συμβολοσειρές δημόσιων κλειδιών των WOTS+, και τα αντίστοιχα ιδιωτικά κλειδιά τους χρησιμοποιούνται για την υπογραφή της ρίζας των δέντρων (HORST) στο επίπεδο κάτω από αυτά.

- Υπάρχει μόνο ένα δέντρο στο επίπεδο $d - 1$ που είναι το ανώτερο δέντρο.
- Υπάρχουν $2^{(d-i-1)*(h/d)}$ δέντρα στο επίπεδο i , $i \in [0, d - 2]$, και η ρίζα του δέντρου στο επίπεδο i θα υπογραφεί από το ιδιωτικό κλειδί του WOTS+ του δέντρου στο επίπεδο $i + 1$.

Αυτό σημαίνει ότι τα δέντρα WOTS+ και HORST δεν εξαρτώνται το ένα από το άλλο, παρόλο που επιλέγονται στην ίδια διαδρομή για την πιστοποίηση υπογραφής. Αυτό το χαρακτηριστικό είναι χρήσιμο για την αποφυγή της ανάγκης προ-υπολογισμού όλων των δέντρων κατά τη δημιουργία κλειδιών, επιτρέποντας έτσι τη διαχείριση ενός μεγάλου αριθμού κρυπτογραφικών κλειδιών υπό ένα μόνο δημόσιο κλειδί SPHINCS.

Αυτή η μεγάλη βάση κρυπτογραφικών κλειδιών πρακτικά καθιστά το SPHINCS ένα σύστημα υπογραφής χωρίς διατήρηση κατάστασης.

Όπως αναφέρθηκε προηγουμένως, το SPHINCS προσδιορίζει μόνο συγκεκριμένες διαδρομές στο υπερδέντρο κατά την υπογραφή ενός μηνύματος. Η διαδρομή παράγεται χρησιμοποιώντας ένα σχήμα διευθυνσιοδότησης για τον εντοπισμό των δημόσιων κλειδιών του WOTS+ στο υπερ-δέντρο. Το σχήμα διευθυνσιοδότησης αποτελείται από το επίπεδο του υπερ-δέντρου, το δέντρο σε αυτό το επίπεδο και το φύλλο εντός αυτού του δέντρου. Χρησιμοποιώντας αυτήν τη μορφή, μπορούμε να προσδιορίσουμε μοναδικά τη θέση κάθε δημόσιου κλειδιού του WOTS+ σε κάθε επίπεδο του υπερ-δέντρου.

Το ιδιωτικό κλειδί SK του SPHINCS περιλαμβάνει:

- Ένα n -bit κλειδί SK_1 που παράγεται χρησιμοποιώντας μια γεννήτρια ψευδοτυχαίων αριθμών. Χρησιμοποιείται για τη δημιουργία τυχαίων σπόρων για τη δημιουργία ιδιωτικών κλειδιών HORST και WOTS+.
- Ένα n -bit κλειδί SK_2 που επίσης παράγεται χρησιμοποιώντας μια γεννήτρια ψευδοτυχαίων αριθμών. Αυτό το κλειδί χρησιμοποιείται για τη δημιουργία μιας σύνοψης σύνοψη, η οποία περιλαμβάνει τόσο έναν τυχαίο δείκτη (π.χ. για την επιλογή κλειδιών ή μονοπατιών) όσο και το ίδιο το μήνυμα, με τρόπο που καθιστά αδύνατη την πρόβλεψή της χωρίς τη γνώση του κλειδιού SK_2 .
- Δυαδικές μάσκες $Q = (Q_0, Q_1, \dots, Q_{p-1})$, οι οποίες χρησιμοποιούνται στα δέντρα HORST, WOTS+, L-δέντρο, και υπερ-δέντρο. Το WOTS+ χρειάζεται $w - 1$ δυαδικές μάσκες, το HORST χρειάζεται $2 \log(t)$ δυαδικές μάσκες, και το L-δέντρο χρειάζεται $2 \lceil \log(l) \rceil$. Συνολικά, η πλήρης δομή SPHINCS χρειάζεται p δυαδικές μάσκες όπου $p = \max(w - 1, 2(h + \lceil \log(l) \rceil), 2 \log(t))$.

Στην εξίσωση $p = \max(w - 1, 2(h + \lceil \log(l) \rceil), 2 \log(t))$, οι παράμετροι l και t σχετίζονται με τις δομές υπογραφής στο SPHINCS:

- Το l αντιπροσωπεύει τον αριθμό των τμημάτων της υπογραφής που χρησιμοποιούνται στα L-δέντρα του συστήματος SPHINCS. Κάθε L-δέντρο είναι ένα Merkle δέντρο, και το l αναφέρεται στον αριθμό των φύλλων ή τμημάτων που συμμετέχουν στη διαδικασία της υπογραφής.
- Το t αναφέρεται στον αριθμό των φύλλων του δέντρου HORST.

Η εξίσωση καθορίζει τον αριθμό των δυαδικών μασκών Q που απαιτούνται για τις διάφορες διαδικασίες υπογραφής στο SPHINCS. Το $w - 1$ αναφέρεται στις μάσκες που χρησιμοποιούνται για τη WOTS+ υπογραφή. Το $2(h + \lceil \log(l) \rceil)$ αναφέρεται στις μάσκες που χρησιμοποιούνται για την υπογραφή στις L-δέντρα. Το $2 \log(t)$ αναφέρεται στις μάσκες που απαιτούνται για τη δομή υπογραφής HORST.

Η συνάρτηση \max επιλέγει τον μέγιστο από αυτούς τους αριθμούς για να καθορίσει τον συνολικό αριθμό των μασκών που απαιτούνται.

Η διεύθυνση των φύλλων των δέντρων στο υψηλότερο επίπεδο, δηλαδή το επίπεδο $d - 1$, δίνεται από:

$$A = (d - 1 \parallel 0 \parallel i)(i \in [2^{h/d} - 1])$$

Παράγονται τον σπόρο $SA \leftarrow F(A, SK_1)$ χρησιμοποιώντας το n-bit μυστικό κλειδί SK_1 . Στη συνέχεια, χρησιμοποιούμε το SA ως τον σπόρο για τη δημιουργία των ιδιωτικών κλειδιών του WOTS+, και υπολογίζουμε τη ρίζα αυτού του δέντρου υψηλού επιπέδου (ας ονομάσουμε τη ρίζα PK_{root}). Το τελικό ιδιωτικό κλειδί και δημόσιο κλειδί του SPHINCS δίνονται από:

$$SK = (SK_1, SK_2, Q)$$

$$PK = (PK_{root}, Q)$$

Για να δημιουργήσουμε μια υπογραφή ενός μηνύματος M με το SPHINCS, πρώτα παράγουμε ένα ζεύγος κλειδιών HORST. Στο δεύτερο βήμα, παράγεται μια συγκεκριμένη διαδρομή του υπερ-δέντρου του SPHINCS. Για την παραγωγή του ζεύγους κλειδιών HORST, πρώτα προσδιορίζεται η διεύθυνση του ζεύγους κλειδιών HORST. Μόλις έχουμε τη διεύθυνση, μπορεί να παραχθεί ο αντίστοιχος τυχαίος σπόρος, και στη συνέχεια, χρησιμοποιώντας τον τυχαίο σπόρο, παράγεται το ζεύγος κλειδιών HORST. Αυτά τα κλειδιά χρησιμοποιούνται για την υπογραφή του μηνύματος. Όλες οι άλλες υπογραφές WOTS+ παράγονται με παρόμιο τρόπο:

προσδιορισμός διεύθυνσης → παραγωγή τυχαίου σπόρου → παραγωγή ζεύγους κλειδιών →_w παραγωγή υπογραφής

Πιο αναλυτικά, η διαδικασία της δημιουργίας υπογραφής περιλαμβάνει τα ακόλουθα βήματα:

- Παράγουμε δύο τυχαίες n -bit τιμές R_1 και R_2 με τη χρήση $F(M, SK_2)$, όπου F συνάρτηση γεννήτριας τυχαίων σπόρων.
- Υπολογίζουμε τη σύνοψη του μηνύματος $D \leftarrow H(R_1, M)$.
- Υπολογίζουμε τη διεύθυνση του HORST $i \leftarrow \text{Chop}(R_2, h)$ και την $\text{Address}_H = (d \parallel i(0, (d-1)h/d) \parallel i((d-1)h/d, h/d))$, όπου Address_H καθορίζει το επίπεδο και τις θέσεις στο υπερ-δέντρο SPHINCS όπου θα παραχθεί το ζεύγος κλειδιών HORST, και $\text{Chop}(M, i)$ η συνάρτηση που επιστρέφει τα πρώτα i bits της συμβολοσειράς M .
- Παράγουμε το ζεύγος κλειδιών HORST και την υπογραφή HORST:
 - Παράγουμε τον σπόρο του ζεύγους κλειδιών HORST με τη χρήση $\text{Seed}_H \leftarrow F(\text{Address}_H, SK_1)$.
 - Παράγουμε το ζεύγος κλειδιών HORST (δημόσιο και ιδιωτικό κλειδί) και στη συνέχεια παράγουμε την υπογραφή HORST S_H εκτελώντας τον αλγόριθμο υπογραφής του HORST με εισόδους (D, Seed_H, Q_H) .
- Παράγουμε όλες τις υπογραφές WOTS+ κατά μήκος της διαδρομής του SPHINCS:
 - Υπολογίζουμε όλες τις διευθύνσεις του WOTS+ στη διαδρομή $\text{Address}_{w,j} = (j \parallel i(0, (d-1-j)h/d) \parallel i((d-1-j)h/d, h/d))$, όπου j είναι το επίπεδο και $j \in [0, d-1]$. Εδώ, η παράμετρος w αναφέρεται στη μεταβλητή του WOTS+ που καθορίζει πόσες φορές εφαρμόζεται η συνάρτηση σύνοψης σε κάθε στοιχείο του ιδιωτικού κλειδιού για την παραγωγή του αντίστοιχου δημόσιου κλειδιού και της υπογραφής. Ένα μεγαλύτερο w μειώνει τον αριθμό των απαιτούμενων λειτουργιών σύνοψης, αλλά αυξάνει το μέγεθος της υπογραφής, επηρεάζοντας έτσι την αποδοτικότητα του συστήματος.
 - Υπολογίζουμε όλους τους σπόρους $\text{Seed}_{w,j} = F(\text{Address}_{w,j}, SK_1)$.
 - Παράγουμε την υπογραφή WOTS+ εκτελώντας τον αλγόριθμο υπογραφής του WOTS+ με εισόδους $(pk_{w,j-1}, \text{Seed}_{w,j}, Q_{WOTS+})$, όπου $pk_{w,j-1}$ είναι η ρίζα του δέντρου του επιπέδου $j-1$. Επίσης, πρέπει να παράξουμε τη διαδρομή πιστοποίησης (Merkle authentication path) auth_{A_j} του αντίστοιχου δημόσιου κλειδιού του WOTS+.

Η υπογραφή του SPHINCS είναι:

$$S_{\text{SPHINCS}} = (i, R_1, S_H, S_{w,0}, \text{auth}_{A_0}, S_{w,1}, \text{auth}_{A_1}, \dots, S_{w,d-1}, \text{auth}_{A_{d-1}})$$

Ο αλγόριθμος επαλήθευσης στο SPHINCS περιλαμβάνει τα εξής βήματα:

- Το πρώτο βήμα περιλαμβάνει τον έλεγχο της υπογραφής HORST. Ο αλγόριθμος επαλήθευσης υπολογίζει τη σύνοψη D υπολογίζοντας $H(R_1, M)$. Στη συνέχεια εκτελείται ο αλγόριθμος επαλήθευσης του HORST με εισόδους (D, Q_{HORST}, S_H) για να ελέγξει την εγκυρότητα της υπογραφής HORST S_H .
- Το δεύτερο βήμα περιλαμβάνει τον έλεγχο όλων των υπογραφών WOTS+. Πρώτα επαληθεύεται η $S_{w,0}$ εκτελώντας τον αλγόριθμο επαλήθευσης του WOTS+ με εισόδους $(pk_H, S_{w,0}, Q_{HORST})$. Στη συνέχεια, επαληθεύονται οι υπογραφές $S_{w,i}$ εκτελώντας τον αλγόριθμο επαλήθευσης του WOTS+ με εισόδους $(pk_{w,i}, S_{w,i}, Q_{WOTS+})$, όπου $i \in [1, d - 1]$.
- Απόρριψη εάν οποιαδήποτε από τις υπογραφές WOTS+ δεν μπορεί να επαληθευτεί.
- Στο υπερ-δέντρο επιπέδου $d - 1$, το βασιζόμενο μέρος αποκτά τη ρίζα του υπερ-δέντρου. Εάν η ρίζα ισούται με το PK_{root} , τότε η υπογραφή $S_{SPHINCS}$ είναι έγκυρη, διαφορετικά απορρίπτεται.

Παρόλο που όλο το σχήμα υπογραφής φαίνεται πολύπλοκο, η ιδέα είναι απλή: το υπερ-δέντρο επιτρέπει μια πολύ μεγάλη βάση κλειδιών χωρίς την ανάγκη προ-υπολογισμού όλων των κλειδιών και των ενδιάμεσων κόμβων.

Πώς το SPHINCS Επιτυγχάνει την Υπογραφή Χωρίς Διατήρηση Κατάστασης: Για την υπογραφή ενός μηνύματος χρησιμοποιώντας το SPHINCS, το πρώτο βήμα περιλαμβάνει την παραγωγή μιας σύνοψης του μηνύματος M . Αυτή η σύνοψη δεν είναι απλώς το αποτέλεσμα μιας συνάρτησης σύνοψης, αλλά τροποποιείται τυχαία με τη χρήση ενός δείκτη, ο οποίος δημιουργείται από μια γεννήτρια ψευδοτυχαίων αριθμών. Ο δείκτης αυτός καθορίζει ποιο σύστημα HORST θα χρησιμοποιηθεί για την υπογραφή της τυχαίας σύνοψης του μηνύματος. Παρόλο που η επιλογή του δείκτη περιέχει τυχαιότητα, η διαδικασία εξακολουθεί να είναι ντετερμινιστική, καθώς εξαρτάται από το ίδιο το μήνυμα. Αυτός ο σχεδιασμός επιτρέπει στο SPHINCS να αποφύγει την ανάγκη διατήρησης κατάστασης, καθιστώντας το ασφαλές για πολλές χρήσεις χωρίς να απαιτείται καταγραφή προηγούμενων υπογραφών.

15.3.1.4 SPHINCS+

Με την κατανόηση του SPHINCS, εισάγουμε τώρα το σύγχρονο σχήμα υπογραφής βασισμένο σε συνόψεις – SPHINCS+ [17]. Μεταξύ των πιο πρόσφατων συστημάτων υπογραφής αυτής της κατηγορίας, το SPHINCS+ έχει σημαντικά πλεονεκτήματα στην ταχύτητα, την ασφάλεια και το μέγεθος της υπογραφής. Γενικά, οι ιδέες του SPHINCS+ και του SPHINCS είναι αρκετά παρόμοιες, αλλά υπάρχουν κάποιες διαφορές:

- Για την υπογραφή μηνυμάτων στο SPHINCS+ χρησιμοποιείται το FORS (Forest Of Random Subsets) αντί του HORST, το οποίο αποτελεί μια βελτίωση του HORST που σχεδιάστηκε από την ομάδα του SPHINCS+. Το FORS αποτελεί ένα σχήμα υπογραφών μερικών χρήσεων. Επιτρέπει τη χρήση πολύ μικρότερων παραμέτρων και έτσι προσφέρει πλεονεκτήματα τόσο σε μέγεθος υπογραφής όσο και σε ταχύτητα.
- Χρησιμοποιείται ένας δημόσια επαληθεύσιμος τρόπος επιλογής του δείκτη φύλλου.

Το δημόσιο κλειδί PK του SPHINCS+ περιλαμβάνει τα εξής:

- Τη ρίζα του υποδέντρου στο υψηλότερο επίπεδο (δηλ. τη ρίζα της δομής υπερ-δέντρου) PK_{root} .
- Έναν τυχαίο σπόρο PK_{seed} .

Το ιδιωτικό κλειδί SK περιλαμβάνει:

- Έναν n-bit σπόρο SK_{seed} που χρησιμοποιείται για τη δημιουργία ιδιωτικών κλειδιών του WOTS+ και του FORS.
- Έναν επίσης n-bit σπόρο SK_{PRF} που χρησιμοποιείται για τη δημιουργία σύνοψης τυχαίου μηνύματος.

Το SPHINCS+ διαφέρει από το αρχικό SPHINCS στις μεθόδους που χρησιμοποιούνται για τον υπολογισμό της σύνοψης μηνύματος και την επιλογή φύλλων:

- Το SPHINCS+ παράγει μια ψευδοτυχαία τιμή R που λειτουργεί ως τυχαίος παράγοντας. Αυτή η τιμή R εξαρτάται από το μήνυμα M που θα υπογραφεί και το SK_{PRF} . Ο υπολογισμός του R γίνεται μη ντετερμινιστικός εισάγοντας μια άλλη παράμετρο τυχαιότητας $OptRand$. Αυτό μπορεί να είναι χρήσιμο για την αποφυγή επιθέσεων πλευρικού καναλιού (side-channel attacks). Έτσι, $R = PRF(SK_{PRF}, OptRand, M)$, που είναι μέρος της υπογραφής.
- Χρησιμοποιώντας το R , παίρνουμε τον δείκτη του φύλλου που θα χρησιμοποιηθεί, καθώς και τη σύνοψη του μηνύματος $(MD \parallel id_x) = H_{msg}(R, PK_{seed}, PK_{root}, M)$, όπου MD σημαίνει σύνοψη μηνύματος και id_x συμβολίζει τον δείκτη φύλλου. Το H_{msg} είναι μια πρόσθετη συνάρτηση σύνοψης με κλειδί για τη συμπίεση του μηνύματος.

Σημειώστε ότι στο SPHINCS+ χρησιμοποιούμε έναν δημόσια επαληθεύσιμο τρόπο για την επιλογή του δείκτη. Αυτό εμποδίζει έναν κακόβουλο παράγοντα από το να επιλέξει έναν τυχαίο φαινομενικά δείκτη και να τον συνδυάσει με ένα μήνυμα της επιλογής του. Δεδομένου ότι ο δείκτης είναι δημόσια επαληθεύσιμος και μπορεί εύκολα να υπολογιστεί από το βασιζόμενο μέρος (relying party), δεν είναι τμήμα της υπογραφής στο SPHINCS+. Η υπογραφή του SPHINCS+ είναι:

$$S_{SPHINCS+} = (MD, R, S_F, S_{w,0}, auth_{A_0}, S_{w,1}, auth_{A_1}, \dots, S_{w,d-1}, auth_{A_{d-1}})$$

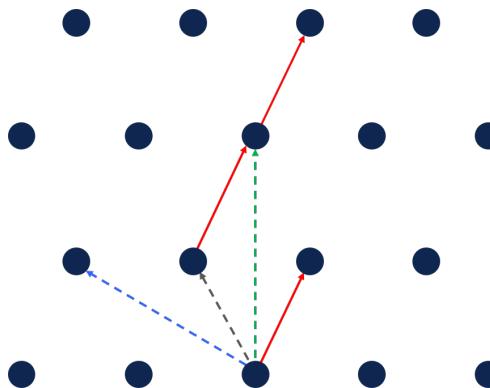
όπου S_F είναι η υπογραφή του FORS, $S_{w,0}, S_{w,1}, \dots, S_{w,d-1}$ οι WOTS+ υπογραφές σε κάθε επίπεδο του υπερδέντρου (hyper-tree), και $auth_{A_0}, auth_{A_1}, \dots, auth_{A_{d-1}}$ οι διαδρομές πιστοποίησης για κάθε επίπεδο του δέντρου. Για να επαληθεύσει μια υπογραφή $S_{SPHINCS+}$, το βασιζόμενο μέρος κάνει τα ακόλουθα:

- Ελέγχει την υπογραφή του FORS πρώτα παράγοντας τα MD και idx . Σημειώστε ότι η διαδικασία παραγωγής των MD και idx είναι ίδια με αυτήν που περιγράφεται στη δημιουργία υπογραφής του SPHINCS+. Στη συνέχεια, χρησιμοποιεί τα MD , idx και S_F ως εισόδους στον αλγόριθμο επαλήθευσης του FORS για να επαληθεύσει το S_F . Για να επαληθεύσει τον δείκτη του υπογράφοντος, το βασιζόμενο μέρος μπορεί επίσης να υπολογίσει το $H_{msg}(R, PK_{seed}, PK_{root}, M)$ και να το συγκρίνει.
- Η υπόλοιπη διαδικασία είναι ίδια με αυτήν στο SPHINCS, όπου το βασιζόμενο μέρος ελέγχει τις υπογραφές WOTS+ κάθε επιπέδου.

15.3.2 Κρυπτογραφία Βασισμένη σε Πλέγματα

Τα κρυπτογραφικά συστήματα βασισμένα σε πλέγματα (Lattice-based Cryptography) αποτελούν μια ολόκληρη κατηγορία συστημάτων που βασίζονται σε δύσκολα προβλήματα σχετικά με χώρους (spaces) που σχηματίζονται από συνδυασμούς συνόλων διανυσμάτων για να σχηματίσουν νέα διανύσματα. Όλα τα νέα διανύσματα που μπορούν να σχηματιστούν από αυτούς τους συνδυασμούς ονομάζονται πλέγμα.

Ένα πλέγμα στην απλή του μορφή αποτελείται από διανύσματα δύο ή τριών διαστάσεων (διανύσματα με 2 ή 3 μεταβλητές). Στην πράξη, τα διανύσματα σε ένα πλέγμα μπορούν να έχουν αυθαίρετο αριθμό διαστάσεων, και ειδικά στην κρυπτογραφία, ο αριθμός των διαστάσεων είναι συνήθως πολύ μεγαλύτερος από τις 2 ή 3 που χρησιμοποιούνται στα απλά παραδείγματα. Στο Σχήμα 15.7 απεικονίζεται ένα παράδειγμα πλέγματος που σχηματίζεται από δύο δισδιάστατα διανύσματα (το ένα αναπαριστάνεται με κόκκινες γραμμές, το άλλο με μαύρες διακεκομένες γραμμές).



Σχήμα 15.7: Πλέγμα που σχηματίζεται από δύο δισδιάστατα διανύσματα.

Κάποια ενδιαφέροντα χαρακτηριστικά των πλεγμάτων είναι ότι μπορεί να σχηματιστεί το ίδιο πλέγμα με διαφορετικά διανύσματα. Για παράδειγμα, τα μπλε και πράσινα διανύσματα (οι διακεκομμένες γραμμές με τελείες) μπορούν να χρησιμοποιηθούν για να σχηματίσουν το ίδιο πλέγμα. Ονομάζουμε το σύνολο των διανυσμάτων που ορίζουν ένα πλέγμα ως την βάση του πλέγματος.

15.3.2.1 Προβλήματα Σχετικά με τη Θεωρία των Πλεγμάτων

Τα προβλήματα που σχετίζονται με τα πλέγματα είναι κεντρικής σημασίας στον τομέα της κρυπτογραφίας και της θεωρίας της πολυπλοκότητας. Ένα πλέγμα είναι ένα μαθηματικό σύνολο σημείων που σχηματίζει ένα κανονικό, επαναλαμβανόμενο μοτίβο στο χώρο. Τα προβλήματα πλέγματος έχουν πολλές πρακτικές εφαρμογές, όπως στην κρυπτογραφία, όπου η ασφάλεια πολλών κρυπτοσυστημάτων βασίζεται στη δυσκολία επίλυσής τους. Τα πιο σημαντικά και μελετημένα προβλήματα περιλαμβάνουν το πρόβλημα του συντομότερου διανύσματος (Shortest Vector Problem – SVP), το πρόβλημα του πλησιέστερου διανύσματος (Closest Vector Problem – CVP) και το πρόβλημα των συντομότερων ανεξάρτητων διανυσμάτων (Shortest Independent Vector Problem – SIVP):

1. Πρόβλημα του Συντομότερου Διανύσματος: Γενικά, είναι δύσκολο να προσδιοριστεί το συντομότερο διάνυσμα σε ένα πλέγμα.
2. Πρόβλημα του Πλησιέστερου Διανύσματος: Δεδομένου ενός διανύσματος, βρείτε το διάνυσμα που είναι πιο κοντά στο δεδομένο διάνυσμα. Με την επίλυση αυτού του προβλήματος μπορεί να επιλυθεί το πρόβλημα του συντομότερου διανύσματος απλώς βρίσκοντας το διάνυσμα που είναι πιο κοντά στο κέντρο. Αντίστροφα, αν λυθεί το πρόβλημα του συντομότερου διανύσματος, μπορεί να γίνει μετασχηματισμός συντεταγμένων και να βρεθεί το πλησιέστερο διάνυσμα σε ένα αυθαίρετο διάνυσμα.
3. Πρόβλημα των Συντομότερων Ανεξάρτητων Διανυσμάτων: Αφορά στην εύρεση της βάσης για ένα πλέγμα με τα συντομότερα δυνατά διανύσματα. Αν λυθεί αυτό, μπορεί να βρεθεί το συντομότερο διάνυσμα βρίσκοντας το συντομότερο διάνυσμα στη βάση του διανύσματος. Αν μπορεί να βρεθεί το συντομότερο διάνυσμα, μπορεί να χρησιμοποιηθεί επαναληπτικά για να βρεθεί το σύνολο των συντομότερων δυνατών διανυσμάτων σε μια βάση.

Αυτά τα προβλήματα έχουν αποδειχθεί ότι είναι δύσκολα προς επίλυση τόσο από κλασικούς όσο και από κβαντικούς υπόλογιστές, καθιστώντας τα ιδιαίτερα σημαντικά για την ανάπτυξη κρυπτογραφικών αλγορίθμων ανθεκτικών σε κβαντικές επιθέσεις. Ωστόσο, η επίλυση ενός από αυτά, μπορεί να οδηγήσει στην επίλυση των υπολοίπων.

Η κρυπτογραφία βασισμένη σε πλέγματα αποτελεί μια ελκυστική οικογένεια κρυπτογραφικών αλγορίθμων για την μετά-κβαντική εποχή. Μπορεί να θεωρηθεί ως μια ειδική περίπτωση κρυπτογραφίας με βάση

το πρόβλημα υποσυνόλων (subset sum problem), το οποίο χρονολογείται από το 1978 με την κρυπτογραφία Knapsack που προτάθηκε από τους Merkle και Hellman [18]. Οι πρώτες προσπάθειες να κατασκευαστούν ασφαλή συστήματα κρυπτογραφίας δημοσίου κλειδιού με βάση το πρόβλημα υποσυνόλων απέτυχαν, ενώ ταυτόχρονα η τεχνική μείωσης πλέγματος [19] έγινε ένα πολύ ισχυρό εργαλείο κρυπτανάλυσης. Σχεδόν δύο δεκαετίες αργότερα, το 1996, ο Ajtai [20] βρήκε έναν σωστό και ασφαλή τρόπο να σχεδιάσει κρυπτογραφικά σχήματα βασισμένα στο πρόβλημα υποσυνόλων. Πιο πρόσφατα, ένα στάδιο ταχείας ανάπτυξης της κρυπτογραφίας βασισμένης σε πλέγματα ξεκίνησε από την πρόταση του Regev το 2005 [21]. Καθώς σχεδιάστηκαν ευέλικτα και ισχυρά κρυπτογραφικά σχήματα με βάση τα προβλήματα πλέγματος, η κρυπτογραφία βασισμένη σε πλέγματα κέρδισε όλο και περισσότερη προσοχή και έγινε ένας ελκυστικός ερευνητικός τομέας της κρυπτογραφίας.

Εκτός από την υποθετική ασφάλεια έναντι κβαντικών επιθέσεων, η κρυπτογραφία βασισμένη σε πλέγματα έχει και πολλά άλλα ελκυστικά χαρακτηριστικά, μερικά από τα οποία περιγράφονται παρακάτω:

- Δυσκολία χειρότερης περίπτωσης (worst-case hardness): Η ασφάλεια των κρυπτογραφικών συστημάτων αποδεικνύεται με βάση τη δυσκολία επίλυσης της μέσης περίπτωσης των σχετικών προβλημάτων (average-case intractability), δηλαδή οι τυχαία επιλεγμένες περιπτώσεις είναι εξίσου δύσκολο να λυθούν με τις πιο δύσκολες περιπτώσεις. Η εργασία του Ajtai [20] και του Regev [21] καθιέρωσε συνδέσεις μεταξύ των προβλημάτων μέσης περίπτωσης (SIS: Short Integer Solution, LWE: Learning with Errors) και των προβλημάτων χειρότερης περίπτωσης (SVP: Shortest Vector Problem, CVP: Closest Vector Problem). Τα αποτελέσματα τους παρέχουν εγγύηση ασφάλειας χειρότερης περίπτωσης στην κρυπτογραφία βασισμένη σε πλέγματα.
- Απλότητα και παράλληλη εκτέλεση: Οι αλγόριθμοι των κρυπτογραφικών συστημάτων βασισμένων σε πλέγματα αποτελείται κυρίως από γραμμικές πράξεις σε πίνακες και πολυώνυμα modulo σχετικά μικρών ακεραίων. Αυτό το είδος πράξεων είναι εξαιρετικά παραλληλίσιμο και μπορεί να επιταχυνθεί με τη χρήση εντολών διανύσματος (vector instructions), προγραμματισμού πολλαπλών νημάτων (multi-thread programming) ή προγραμματισμού πολλαπλών πυρήνων (multi-core programming).
- Ευελιξία στην κατασκευή ισχυρών κρυπτογραφικών συστημάτων: Το πρόβλημα LWE (Learning-with-Errors) που προτάθηκε αρχικά από τον Regev [21], ήταν εξαιρετικά ευέλικτο στην κατασκευή πολλών ειδών κρυπτογραφικών λύσεων, όπως σχήματα κρυπτογράφησης δημόσιου κλειδιού, σχήματα ψηφιακής υπογραφής, σχήματα κρυπτογράφησης με βάση την ταυτότητα και πλήρως ομοιομορφικά σχήματα κρυπτογράφησης.

15.3.3 Κρυπτογραφία Βασισμένη σε Κώδικα

Η κρυπτογραφία βασισμένη σε κώδικα (Code-based Cryptography) χρησιμοποιεί τη θεωρία των κωδικών διόρθωσης σφαλμάτων (error-correcting codes) [22, 23]. Η ασφάλεια αυτών των συστημάτων βασίζεται στη δυσκολία αποκωδικοποίησης ορισμένων τύπων κωδικών διόρθωσης σφαλμάτων. Πιο συγκεκριμένα, όταν τα δεδομένα κρυπτογραφούνται χρησιμοποιώντας ένα τέτοιο σύστημα, κωδικοποιούνται με την προσθήκη πλεονασμού σε αυτά τα δεδομένα, επιτρέποντας τον εντοπισμό και τη διόρθωση σφαλμάτων. Ωστόσο, η κωδικοποίηση αυτή μετατρέπει τα δεδομένα σε μια μη κατανοητή και μη επεξεργάσιμη ακολουθία για όποιον προσπαθεί να τα υποκλέψει και να τα αποκρυπτογραφήσει χωρίς το κλειδί αποκρυπτογράφησης.

Η διαδικασία αποκωδικοποίησης, που οδηγεί στην αποκρυπτογράφηση των δεδομένων, είναι σχεδιασμένη να είναι ιδιαίτερα δύσκολη, ακόμη και για ισχυρούς υπολογιστές. Η δυσκολία προκύπτει από το γεγονός ότι η αποκωδικοποίηση ενός τυχαία επιλεγμένου γραμμικού κώδικα θεωρείται ένα υπολογιστικά δύσκολο πρόβλημα (NP-hard) [24].

Τα συστήματα κρυπτογράφησης που βασίζονται σε κώδικα προέρχονται από την πρόταση του McEliece το 1978 [22] και είναι από τα πιο μελετημένα μετα-κβαντικά συστήματα. Ανήκουν στα παλαιότερα συστήματα δημόσιου κλειδιού, ωστόσο, σε αντίθεση με τον RSA, που έχει περίπου την ίδια ηλικία, και την κρυπτογραφία

ελλειπτικών καμπυλών (Elliptic Curve Cryptography – ECC), που είναι περίπου 10 χρόνια νεότερη, δεν έχουν υιοθετηθεί ευρέως, παρά το μεγάλο ακαδημαϊκό ενδιαφέρον. Ταυτόχρονα, το ιστορικό ασφάλειας της κρυπτογραφίας βασισμένης σε κώδικα είναι πολύ ισχυρότερο από αυτό του RSA. Οι παράμετροι που πρότεινε ο McEliece το 1978 για ασφάλεια 64-bit έσπασαν το 2008, μετά από περίπου 2^{64} υπολογισμούς [25]. Περισσότερα από 30 χρόνια έρευνας δεν άλλαξαν σημαντικά το κόστος της επίθεσης, ενώ η ασφάλεια του RSA μειώθηκε από εκθετική² σε υποεκθετική, καθώς αυξάνεται με ρυθμό μικρότερο από τον εκθετικό, λόγω της ανάπτυξης πιο αποδοτικών αλγορίθμων επίθεσης.

Το κύριο μειονέκτημα του αρχικού συστήματος McEliece είναι το μεγάλο μέγεθος κλειδιού. Έρευνες για βελτιστοποίησεις έχουν οδηγήσει σε σημαντικά μικρότερα ιδιωτικά κλειδιά και ελαφρώς μικρότερα δημόσια κλειδιά, διατηρώντας την ασφάλεια του συστήματος. Σύγχρονες παραλλαγές, όπως αυτές που χρησιμοποιούν κώδικες QC-LDPC (quasi-cyclic codes: low density parity check codes) και QC-MDPC (quasi-cyclic codes: moderate density parity check codes), έχουν επιτύχει μείωση του μεγέθους των δημόσιων κλειδιών διατηρώντας υψηλό επίπεδο ασφάλειας. Αυτές οι βελτιστοποίησεις περιλαμβάνουν τη χρήση κωδικών χαμηλής πυκνότητας παρακολούθησης και άλλων ειδικών κωδικών που βοηθούν στη μείωση της πολυπλοκότητας και του μεγέθους των κλειδιών [27].

Τα συστήματα που βασίζονται σε κώδικα και κρίνονται κατάλληλα περιλαμβάνουν τα Classic McEliece [22], BIKE (Bit Flipping Key Encapsulation) [28] και HQC (Hamming Quasi-Cyclic) [29]. Το Classic McEliece ήταν ο πρώτος φιναλίστ που επιλέχθηκε από το NIST για συστήματα κρυπτογράφησης, ενώ τα BIKE και HQC επιλέχθηκαν ως αναπληρωματικοί υποψήφιοι. Τα δύο τελευταία χρησιμοποιούν εξειδικευμένους κώδικες προκειμένου να μειώσουν το μέγεθος του δημόσιου κλειδιού, το οποίο θεωρείται το κύριο μειονέκτημα των συστημάτων που βασίζονται σε κώδικα.

15.3.4 Κρυπτογραφία Βασισμένη σε Ισογένειες

Μια ισογένεια (isogeny) μεταξύ ελλειπτικών καμπυλών είναι μια αλγεβρική απεικόνιση που διατηρεί τη δομή της ομάδας των σημείων στις καμπύλες. Συγκεκριμένα, μια ισογένεια ϕ μεταξύ δύο ελλειπτικών καμπυλών E_1 και E_2 ικανοποιεί την ιδιότητα $\phi(P+Q) = \phi(P) + \phi(Q)$ για όλα τα σημεία P και Q στην E_1 . Με άλλα λόγια, η ισογένεια «μετασχηματίζει» τα σημεία της καμπύλης E_1 στα αντίστοιχα σημεία της καμπύλης E_2 , δηλαδή παίρνει κάθε σημείο από την καμπύλη E_1 και το απεικονίζει σε ένα συγκεκριμένο σημείο της καμπύλης E_2 , διατηρώντας την ίδια αλγεβρική σχέση μεταξύ τους (αυτό σημαίνει ότι οι αλγεβρικές πράξεις, όπως το άθροισμα σημείων, παραμένουν συνεπείς πριν και μετά τον μετασχηματισμό). Επιπλέον, μια ισογένεια διατηρεί το ειδικό σημείο στο άπειρο (το ουδέτερο στοιχείο της ομάδας), μεταφέροντας το σημείο στο άπειρο της E_1 στο σημείο στο άπειρο της E_2 .

Η κρυπτογραφία βασισμένη σε ισογένειες (isogeny-based cryptography) χρησιμοποιεί τις ισογένειες μεταξύ ελλειπτικών καμπυλών σε πεπερασμένα πεδία. Συγκεκριμένα, χρησιμοποιεί απεικονίσεις μεταξύ ελλειπτικών καμπυλών για την κατασκευή κρυπτοσυστημάτων δημοσίου κλειδιού. Το βασικό πρόβλημα της ισογένειας είναι να βρεθεί μια ισογένεια μεταξύ δύο ελλειπτικών καμπυλών που είναι γνωστό ότι είναι ισογένειες [30]. Η χρήση ισογενειών σε κρυπτογραφικά πρωτόκολλα εξαρτάται από το αν ο βαθμός της ισογένειας είναι γνωστός ή μυστικός και από το αν υπάρχουν επιπλέον πληροφορίες.

Η κρυπτογραφία βασισμένη σε ισογένειες (isogeny-based cryptography) θεωρήθηκε πως προσέφερε υποσχόμενες εναλλακτικές λύσεις στα παραδοσιακά κρυπτογραφικά συστήματα, ιδιαίτερα λόγω της ανθεκτικότητάς της σε επιθέσεις από κβαντικούς υπολογιστές. Οι υπεριδιάζουσες (supersingular) ελλειπτικές καμπύλες, οι οποίες είναι ελλειπτικές καμπύλες με συγκεκριμένες μαθηματικές ιδιότητες σε πεπερασμένα πεδία, πά-

²Η εκθετική ασφάλεια σε κρυπτογραφικά συστήματα αναφέρεται στην ιδιότητα αυτών των συστημάτων να αυξάνουν την ασφάλεια τους εκθετικά με το μέγεθος του κλειδιού ή των παραμέτρων τους. Αυτό σημαίνει ότι ο αριθμός των υπολογιστικών πόρων που απαιτούνται για να σπάσει το σύστημα αυξάνεται εκθετικά καθώς το μήκος του κλειδιού αυξάνεται. Για παράδειγμα, αν ένα πρόβλημα έχει εκθετική ασφάλεια, η δυσκολία επίλυσής του διπλασιάζεται κάθε φορά που το μήκος του κλειδιού αυξάνεται κατά ένα bit. Αυτό καθιστά την επίλυση του προβλήματος πρακτικά αδύνατη για μεγάλες παραμέτρους, ακόμη και με την πρόοδο της υπολογιστικής ισχύος, συμπεριλαμβανομένων των κβαντικών υπολογιστών [26].

ζουν κεντρικό ρόλο. Η ασφάλεια αυτών των συστημάτων βασίζεται στη δυσκολία υπολογισμού ισογενειών μεταξύ υπεριδιαζουσών ελλειπτικών καμπυλών. Τα κύρια κρυπτογραφικά πρωτόκολλα με βάση τις ισογένειες περιλαμβάνουν πρωτόκολλα ανταλλαγής κλειδιών και συστήματα ψηφιακών υπογραφών.

Το πρώτο σύστημα βασισμένο σε ισογένειες χρονολογείται από το 1997, αλλά οι πρώτες δημόσια προσβάσιμες προτάσεις από τους Couveignes [31] και Rostovtsev-Stolbunov [32] είναι από το 2006. Μέχρι τότε, οι ισογένειες είχαν χρησιμοποιηθεί ως εργαλείο επίθεσης [33, 34] το 2002 και 2005, ενώ οι Charles, Goren, και Lauter είχαν σχεδιάσει μια συνάρτηση σύνοψης βασισμένη σε ισογένειες [35]. Τα συστήματα των Couveignes και Rostovtsev-Stolbunov χρησιμοποιούν ισογένειες μεταξύ κανονικών ελλειπτικών καμπυλών σε πεπερασμένα πεδία για τη δημιουργία ενός συστήματος ανταλλαγής κλειδιών, που αναφέρεται ως CRS. Η επίθεση του Shor [1], που σπάει την κρυπτογραφία ελλειπτικών καμπυλών βασισμένη στο πρόβλημα διακριτού λογαρίθμου, δεν επηρεάζει αυτές τις κατασκευές, αλλά το 2010 οι Childs, Jao, και Soukharev [36] έδειξαν ότι το CRS μπορεί να σπάσει με μια υποεκθετική κβαντική επίθεση λόγω του Kuperberg [37]. Αυτό σημαίνει ότι οι παράμετροι του συστήματος CRS πρέπει να αυξήθουν ασυμπτωτικά, κάνοντάς το ήδη αργό σύστημα ακόμη πιο αργό, αλλά δεν σημαίνει ότι το σύστημα είναι πλήρως ανασφαλές.

Το 2011, οι Jao και De Feo [38] σχεδίασαν ένα διαφορετικό σύστημα βασισμένο σε ισογένειες που χρησιμοποιεί ισογένειες μεταξύ υπεριδιαζουσών καμπυλών σε επεκτάσεις πεδίων και δεν έχει την ίδια αδυναμία με το προαναφερόμενο σύστημα CRS. Σύμφωνα με ότι γνωρίζουμε σήμερα, προσφέρει εκθετική ασφάλεια, ακόμη και έναντι κβαντικών επιθέσεων. Ένα μικρό μειονέκτημα σε σχέση με το CRS είναι η πιο περίπλοκη ροή δεδομένων, καθώς και η αλλαγή της υπόθεσης ασφάλειας. Συγκεκριμένα, αντί να βασίζεται στο πρόβλημα καθαρής εύρεσης ισογενειών, το νέο σύστημα βασίζεται σε ένα πρόβλημα όπου ο αντίταλος έχει πρόσβαση σε επιπλέον πληροφορίες σχετικά με τις ισογένειες, γεγονός που μπορεί να επηρεάσει την ασφάλεια σε συγκεκριμένες περιπτώσεις. Το σύστημα χρησιμοποιεί καμπύλες παρόμοιες με τη συνάρτηση κατακερματισμού των Charles-Goren-Lauter, αλλά χρειάζεται περισσότερες υποθέσεις ασφαλείας και δύο τύπους ισογενειών αντί για έναν.

Τα τελευταία χρόνια, η έρευνα στα συστήματα βασισμένα σε ισογένειες επικεντρώθηκε κυρίως στο πρωτόκολλο Supersingular Isogeny Diffie-Hellman (SIDH), ένα από τα πιο γνωστά πρωτόκολλα το οποίο επιτρέπει σε δύο μέρη να δημιουργήσουν ένα κοινό μυστικό κλειδί μέσω ενός μη ασφαλούς καναλιού επικοινωνίας, διατηρώντας την ασφάλεια μέσω της χρήσης ισογενειών. Πρόσφατα ευρήματα ωστόσο έχουν δείξει ότι τόσο το SIKE όσο και το SIDH είναι ευάλωτα σε αποτελεσματικές επιθέσεις ανάκτησης κλειδιών. Κατά συνέπεια, δεν θεωρούνται πλέον ασφαλή για κρυπτογραφικές εφαρμογές. Έτσι, η ομάδα του SIKE αποφάσισε να αποσύρει την υποψηφιότητα της από τον διαγωνισμό του NIST³.

15.3.5 Πολυμεταβλητή Κρυπτογραφία

Η πολυμεταβλητή κρυπτογραφία (Multivariate cryptography) ξεκίνησε στα τέλη της δεκαετίας του 1980 και βασίζεται στη δυσκολία επίλυσης ενός συστήματος πολυμεταβλητών τετραγωνικών εξισώσεων σε πεπερασμένα πεδία. Είναι δυνατό να δημιουργηθούν σχήματα υπογραφών από συστήματα εξισώσεων με ομοιόμορφα τυχαίους συντελεστές, και αυτά θεωρούνται τα πιο ασφαλή πολυμεταβλητά συστήματα. Ωστόσο, τα πιο αποδοτικά σχήματα χρησιμοποιούν συστήματα εξισώσεων με «παγίδες» (trapdoors), τα οποία φαίνονται τυχαία στους εξωτερικούς παρατηρητές, αλλά περιέχουν κάποια κρυφή δομή που είναι γνωστή μόνο στο άτομο που τα κατασκεύασε. Χάρη σε αυτές τις δομές, είναι δυνατό να βρεθούν οι λύσεις του συστήματος εξισώσεων πολύ πιο αποτελεσματικά από ότι θα ήταν χωρίς τη γνώση της κρυφής δομής.

Επί του παρόντος, τα συστήματα κρυπτογράφησης πολλαπλών μεταβλητών δεν είναι πολύ αποτελεσματικά, συχνά με πολύ μεγάλα δημόσια κλειδιά και μεγάλους χρόνους αποκρυπτογράφησης. Αναφορικά με τις υπογραφές ωστόσο, τα πράγματα φαίνονται κάπως καλύτερα. Από τα δεκαεννέα συστήματα υπογραφών που υποβλήθηκαν στον διαγωνισμό NIST Post-Quantum Cryptography (PQC), τα επτά ήταν συστήματα υπογραφών πολλαπλών μεταβλητών. Δύο από αυτά τα επτά συστήματα προχώρησαν στον τρίτο γύρο της

³<https://csrc.nist.gov/csrc/media/Projects/post-quantum-cryptography/documents/round-4-submissions/sike-team-note-insecure.pdf>

διαδικασίας NIST PQC. Το σύστημα Rainbow [39] σχεδιάστηκε το 2004 από τους Jintai Ding και Dieter Schmidt και βασίζεται στο σχέδιο υπογραφής Oil-Winegar που εφευρέθηκε από τον Jacques Patarin. Το σύστημα GeMMS (Great Multivariate Short Signature) είναι ένα σύστημα υπογραφών βασισμένο σε πολλές μεταβλητές που παράγει μικρές υπογραφές. Διαθέτει γρήγορη διαδικασία επαλήθευσης και μεσαίο/μεγάλο δημόσιο κλειδί. Επιλέχθηκε ως «αναπληρωματικός υποψήφιος».

Αυτά τα συστήματα έχουν πολύ μικρά μεγέθη υπογραφής (μέχρι 33 bytes), αλλά έχουν αρκετά μεγάλα δημόσια κλειδιά (160 KB ή περισσότερα).

15.4 Μετάβαση στην Μετα-Κβαντική Εποχή

Σύμφωνα με τον ENISA [40], εάν κάποια κρυπτογραφημένα δεδομένα πρέπει να διατηρηθούν εμπιστευτικά για περισσότερα από 10 χρόνια και ο κίνδυνος κάποιος επιτιθέμενος να αποκτήσει πρόσβαση σε ένα κρυπτογραφημένο κείμενο θεωρείται υψηλός, τότε πρέπει να γίνουν άμεσες διορθωτικές ενέργειες αναφορικά με τον τρόπο με τον οποίο αυτά κρυπτογραφούνται. Διαφορετικά, η εμπιστευτικότητά τους θα τεθεί σε κίνδυνο μόλις ο επιτιθέμενος αποκτήσει πρόσβαση σε έναν μεγάλο κβαντικό υπολογιστή. Δεδομένου ότι η διαδικασία του NIST για την επιλογή των κατάλληλων μετα-κβαντικών αλγορίθμων, θα συνεχίσει να λειτουργεί για μερικά χρόνια, υπάρχουν ουσιαστικά δύο βιώσιμες επιλογές για τον χειρισμό αυτού του προβλήματος:

- Η πρώτη επιλογή είναι να γίνει άμεσα μετάβαση στις λεγόμενες υβριδικές υλοποιήσεις που χρησιμοποιούν έναν συνδυασμό προ-κβαντικών (σύγχρονων συμβατικών) και μετα-κβαντικών συστημάτων.
- Η δεύτερη επιλογή είναι να χρησιμοποιηθεί το εννοιολογικά εύκολο, αλλά οργανωτικά πολύπλοκο μέτρο ανάμειξης προ-κοινοποιημένων κλειδιών, για όλα τα κλειδιά που έχουν δημιουργηθεί μέσω κρυπτογραφίας δημόσιου κλειδιού.

Οι δύο αυτές επιλογές αναφέρονται λεπτομερώς στη συνέχεια.

Για συστήματα που βασίζονται στην κρυπτογραφία δημοσίου κλειδιού, όπως αυτά που κάνουν εκτενή χρήση ψηφιακών υπογραφών, και τα οποία θεωρείται δύσκολο να αναβαθμιστούν στο μέλλον, είναι σκόπιμο να ενσωματωθεί από τώρα ένα μετα-κβαντικό σύστημα. Με αυτόν τον τρόπο, διασφαλίζεται η ασφαλής λειτουργία της υπηρεσίας, ακόμα και όταν οι κβαντικοί υπολογιστές γίνουν διαθέσιμοι.

15.4.1 Υβριδικά Συστήματα

Τα τελευταία χρόνια τα μετα-κβαντικά συστήματα μελετώνται εκτενώς και νέα γνώση προστίθεται διαρκώς αναφορικά με την ανθεκτικότητά τους απέναντι σε επιθέσεις. Για αυτά τα συστήματα δεν υπάρχει καμία εγγύηση ότι είναι και θα παραμείνουν ασφαλή. Οι κρυπταναλυτές που τα μελετούν ενδέχεται να μην έχουν λάβει υπόψη τους ή να μην έχουν ανακαλύψει σημαντικές επιθέσεις, οι οποίες μπορεί να είναι εφικτές και αποτελεσματικές ακόμα και με τα σημερινά συστήματα. Επιπλέον, σφάλματα στο περίπλοκο νέο οικοσύστημα του μετα-κβαντικού λογισμικού κρυπτογράφησης μπορεί να οδηγήσουν σε περαιτέρω προβλήματα ασφάλειας. Τα εργαλεία τελευταίας τεχνολογίας μπορούν να εγγυηθούν μερικώς την ορθή λειτουργία του κρυπτογραφικού λογισμικού, ότι, για παράδειγμα, δεν παράγει υπερχειλίσεις buffer, αλλά η επίσημη επαλήθευση της πλήρους λειτουργικής ορθότητας παραμένει πρόκληση. Κατά συνέπεια, υπάρχει ο κίνδυνος η μετάβαση από την κρυπτογραφία προ-κβαντικής κρυπτογραφίας σε ένα μετα-κβαντικό κρυπτοσύστημα να βλάψει την ασφάλεια, όχι μόνο να μην προστατεύσει από τους κβαντικούς υπολογιστές αλλά και να το καταστήσει ευάλωτο και στους σημερινούς υπολογιστές.

Για να αντιμετωπιστεί αυτός ο κίνδυνος, συνιστάται η μετα-κβαντική κρυπτογραφία να αποτελέσει ένα επιπλέον επίπεδο μαζί με την συμβατική ή προ-κβαντική κρυπτογραφία και όχι να χρησιμοποιηθεί ως αντικαταστάτης αυτής. Με αυτόν τον τρόπο δημιουργούνται τα υβριδικά κρυπτοσυστήματα ή διπλά κρυπτοσυστήματα καθώς γίνεται ταυτόχρονη χρήση συμβατικών αλγορίθμων με μετα-κβαντικούς αλγορίθμους για

την προστασία των δεδομένων. Ενδεικτικά, η χρήση υβριδικών ή διπλών κρυπτοσυστημάτων απαιτεί τις ακόλουθες ενέργειες:

- Στην περίπτωση των ψηφιακών υπογραφών, υπογραφή με ένα προ-κβαντικό κρυπτοσύστημα και με ένα μετα-κβαντικό κρυπτοσύστημα. Το βασιζόμενο μέρος (relying party) μπορεί να ελέγχει και τις δύο υπογραφές.
- Στην περίπτωση της κρυπτογράφησης, κρυπτογράφηση αρχικά με ένα συμβατικό/προ-κβαντικό κρυπτοσύστημα και, στη συνέχεια, κρυπτογράφηση του αποτελέσματος με έναν μετα-κβαντικό αλγόριθμο, έτσι ώστε η αποκρυπτογράφηση να απαιτεί γνώση και των δύο κλειδιών αποκρυπτογράφησης.

Για παράδειγμα, ο μηχανισμός ανταλλαγής κλειδιών CECPQ1 (συνδυασμένη ελλειπτική καμπύλη + μετα-κβαντική) της Google, συνδυάζει ένα συμβατικό κρυπτοσύστημα ελλειπτικών καμπυλών, το X25519, με ένα μετα-κβαντικό κρυπτοσύστημα που βασίζεται σε πλέγματα, το NewHope-1024. Το CECPQ2a συνδυάζει το X25519 με ένα άλλο μετα-κβαντικό κρυπτοσύστημα που βασίζεται σε πλέγματα, το NTRU-HRSS-701. Το CECPQ2b συνδυάζει το X25519 με ένα επίσης μετα-κβαντικό κρυπτοσύστημα που βασίζεται σε ισογένειες, το SIKE-p434. Σε κάθε περίπτωση, ακόμη και αν το μετα-κβαντικό κρυπτοσύστημα αποτύχει εντελώς, τα δεδομένα του χρήστη προστατεύονται από το X25519. Κάποιοι από αυτούς τους μετα-κβαντικούς αλγορίθμους μπορεί να αποδειχθεί ότι δεν είναι ασφαλείς, ακόμη και με τους σημερινούς υπολογιστές, οπότε η χρήση του κρυπτοσυστήματος ελλειπτικών καμπυλών θα εξακολουθεί να παρέχει την καλύτερη ασφάλεια που μπορεί να προσφέρει η σημερινή τεχνολογία.

Ένα καλά σχεδιασμένο διπλό κρυπτοσύστημα θεωρείται ισχυρό εάν τουλάχιστον ένα από τα συστατικά κρυπτοσυστήματα είναι ισχυρό. Αυτό σημαίνει, ότι η προσθήκη μετά-κβαντικής κρυπτογραφίας δεν βλάπτει την συμβατική (προ-κβαντική) ασφάλεια που παρέχεται σήμερα από τους ισχυρούς αλγορίθμους. Αυτό παρέχει επίσης έναν απλό τρόπο ενσωμάτωσης της μετά-κβαντικής κρυπτογραφίας ενώ το σύστημα παραμένει συμμορφωμένο με τους υπάρχοντες κανονισμούς και πολιτικές που απαιτούν τη χρήση ισχυρών αλγορίθμων. Έτσι, τα διπλά κρυπτοσυστήματα διευκολύνουν την ταχεία ανάπτυξη της μετά-κβαντικής κρυπτογραφίας.

15.4.2 Μέτρα Προστασίας για Προ-Κβαντική Κρυπτογραφία

Οι χρήστες που δεν θέλουν να ξεκινήσουν την ανάπτυξη μετα-κβαντικών συστημάτων προτού τυποποιηθούν, αλλά ανησυχούν για τη μακροπρόθεσμη εμπιστευτικότητα των μεταδιδόμενων δεδομένων τους, μπορούν να προστατεύσουν τα συστήματά τους συμπεριλαμβάνοντας πρόσθετο κρυπτογραφικό υλικό στη διαδικασία δημιουργίας κρυπτογραφικών κλειδιών και αποφεύγοντας τη χρησιμοποίηση κλειδιών που δημιουργούνται αποκλειστικά με τη χρήση κρυπτογραφίας δημοσίου κλειδιού ή βασίζονται σε αυτή, π.χ. εδραίωση κλειδιών με τη χρήση του Diffie-Hellman ή μεταφορά κλειδιού με τη χρήση RSA. Σε αυτή την περίπτωση, το πρόσθετο κρυπτογραφικό υλικό θα πρέπει να διατηρείται από όλους τους συμμετέχοντες, επιπλέον του κρυπτογραφικού υλικού που παράγεται μέσω των μεθόδων δημοσίου κλειδιού. Ωστόσο, αυτό συνεπάγεται την ανάγκη διατήρησης κοινών δεδομένων για κάθε ζεύγος συμμετεχόντων, γεγονός που καθιστά αυτή την προσέγγιση κατάλληλη μόνο για συστήματα που μπορούν να διατηρούν κατάσταση και έχουν περιορισμένο αριθμό συμμετεχόντων.

Το Z RTP [41], ένα πρωτόκολλο συμφωνίας κλειδιών που βασίζεται στο Diffie-Hellman, περιλαμβάνει έναν τέτοιο μηχανισμό που ονομάζεται «συνέχεια κλειδιού» (key continuity) ως μέτρο κατά των επιθέσεων “man-in-the-middle” (MITM). Το πρωτόκολλο Z RTP, που προσδιορίστηκε το 2006, δεν αναφέρει την ασφάλεια στη μετα-κβαντική εποχή ως κίνητρο, αλλά είναι η πρώτη περιγραφή αυτής της ιδέας. Προχωρά επίσης περισσότερο από άλλα πρωτόκολλα στην ενημέρωση των κοινόχρηστων μυστικών δεδομένων. Το πιο πρόσφατο πρωτόκολλο Wireguard [42] χρησιμοποιεί ένα κοινόχρηστο προκαθορισμένο κλειδί (Pre-shared Key – PSK) και το περιλαμβάνει στην παραγωγή κλειδιών συνόδου, αλλά δεν αλλάζει το PSK. Το Wireguard βασίζεται στο Noise PSK [43]. Το Wireguard αναφέρει ρητά το PSK ως μέτρο προστασίας στην μετα-κβαντική εποχή. Στο [44] παρουσιάζεται μια μικρή προσαρμογή για την επίτευξη καλύτερης προστασίας σε αυτό το σενάριο, ενώ στο [45] μια πλήρως μετα-κβαντική έκδοση.

Η ακόλουθη περιγραφή ακολουθεί την προσέγγιση του Z RTP στο ότι τα κοινά μυστικά δεδομένα αλλάζουν με κάθε χρήση του δημόσιου κλειδιού με σύνοψη νέων δεδομένων. Η συμπεριληφθη μυστικών δεδομένων σε χρήσεις δημόσιου κλειδιού διασφαλίζει την προώθηση μυστικότητας (forward secrecy). Η αλλαγή του διατηρημένου κοινόχρηστου μυστικού κατά τη διάρκεια κάθε χρήσης του δημόσιου κλειδιού με μια συνάρτηση σύνοψης, διασφαλίζει ότι μια παραβίαση του συστήματος μεταγενέστερα, δεν μπορεί να ανακτήσει τα προηγούμενα κλειδιά συνόδου από το διατηρημένο κοινόχρηστο μυστικό και τα καταγεγραμμένα δεδομένα σύνδεσης, ακόμα κι αν ο εισβολέας έχει κβαντικό υπολογιστή και μπορεί έτσι να σπάσει την προ-κβαντική κρυπτογράφηση δημόσιου κλειδιού.

Έστω r η διατηρημένη κοινή μυστική τιμή PSK και s τα νέα κοινόχρηστα δεδομένα, που λαμβάνονται από μια λειτουργία δημόσιου κλειδιού. Τα προαναφερθέντα πρωτόκολλα βασίζονται στην ανταλλαγή κλειδιών Diffie-Hellman, αλλά αυτή η προσέγγιση μπορεί επίσης να χρησιμοποιηθεί για πρωτόκολλα που βασίζονται στον RSA. Κάθε φορά που το αρχικό πρωτόκολλο καλεί μια συνάρτηση για τη δημιουργία του κλειδιού συνόδου k , οι είσοδοι στη συνάρτηση δημιουργίας κλειδιού (KDF) θα πρέπει να επεκτείνονται ώστε να περιλαμβάνουν το r :

$$k = KDF(s, \text{"κλειδί συνόδου"}, r, *)$$

όπου $*$ μπορεί να περιλαμβάνει κάποια δεδομένα περιβάλλοντος (μηνύματα χειραψίας, δημόσια κλειδιά, συμβολοσειρές αναγνωριστικού, κτλ.). Αυτό διασφαλίζει ότι ένας επιτιθέμενος μπορεί να ανακτήσει το k μόνο εάν έχει αποκτήσει το r καθώς και το s .

Μετά τον υπολογισμό του k , το διατηρημένο μυστικό θα πρέπει να ενημερωθεί σε:

$$r' = KDF(k, r)$$

πιθανώς συμπεριλαμβάνοντας άλλα δεδομένα περιβάλλοντος στα ορίσματα της KDF.

Αξίζει να σημειωθεί πως το πρωτόκολλο πρέπει να επαληθεύσει ότι και τα δύο μέρη γνωρίζουν την τιμή s πριν αντικαταστήσουν το r . Το Z RTP [41] παρέχει μια λύση που χρησιμοποιεί δύο μεταβλητές για διατηρούμενες μυστικές τιμές προκειμένου να αποφευχθεί ο αποσυγχρονισμός.

Η παραπάνω περιγραφή αφήνει ανοιχτό τον τρόπο με τον οποίο οι χρήστες έλαβαν την πρώτη τιμή PSK r . Οι χρήστες που ανησυχούν για τη μακροπρόθεσμη ασφάλεια θα πρέπει να κανονίσουν να μοιράζονται τέτοια κλειδιά μέσω άλλων καναλιών (σαρωμένος κωδικός QR, συνθηματικό, κ.α.). Σε σενάρια με προκαθορισμένα μοτίβα επικοινωνίας, όπως ένας κύριος διακομιστής που επικοινωνεί με απομακρυσμένες καταχωριμένες συσκευές, το PSK μπορεί να παρέχεται μαζί με τις συσκευές. Κάθε συσκευή θα πρέπει να λαμβάνει ένα μοναδικό PSK που είναι γνωστό μόνο στη συσκευή και στο διακομιστή.

Βιβλιογραφία

- [1] Peter W. Shor. “Algorithms for quantum computation: discrete logarithms and factoring”. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. Santa Fe, NM, USA: IEEE Comput. Soc. Press, 1994, pp. 124–134. ISBN: 978-0-8186-6580-6. doi: 10.1109/SFCS.1994.365700.
- [2] Frank Arute et al. “Quantum supremacy using a programmable superconducting processor”. In: *Nature* 574.7779 (Oct. 2019), pp. S05–S10. ISSN: 1476-4687. doi: 10.1038/s41586-019-1666-5.
- [3] Lov K. Grover. “A fast quantum mechanical algorithm for database search”. en. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing – STOC ’96*. Philadelphia, Pennsylvania, United States: ACM Press, 1996, pp. 212–219. ISBN: 978-0-89791-785-8. doi: 10.1145/237814.237866.

- [4] Ralph C. Merkle. "A Certified Digital Signature". In: *Advances in Cryptology — CRYPTO' 89 Proceedings*. Ed. by Gilles Brassard. New York, NY: Springer New York, 1990, pp. 218–238. ISBN: 978-0-387-34805-6.
- [5] Johannes Buchmann, Erik Dahmen, and Michael Szydlo. "Hash-based Digital Signature Schemes". en. In: *Post-Quantum Cryptography*. Ed. by Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 35–93. ISBN: 978-3-540-88701-0 978-3-540-88702-7. doi: 10.1007/978-3-540-88702-7_3.
- [6] Ralph C. Merkle. "Secrecy, Authentication and Public Key Systems". PhD thesis. Stanford University, 1979.
- [7] Whitfield Diffie and Martin E. Hellman. "New directions in cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. doi: 10.1109/TIT.1976.1055638.
- [8] Daniel J. Bernstein et al. "SPHINCS: Practical Stateless Hash-Based Signatures". In: *Advances in Cryptology – EUROCRYPT 2015*. Ed. by Elisabeth Oswald and Marc Fischlin. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 368–397. ISBN: 978-3-662-46800-5.
- [9] Andreas Helsing, Denis Butin, Stefan-Lukas Gazdag, Joost Rijneveld, and Aziz Mohaisen. XMSS: eXtended Merkle Signature Scheme. RFC 8391. May 2018. doi: 10.17487/RFC8391.
- [10] David A. Cooper et al. *Recommendation for Stateful Hash-Based Signature Schemes*. Tech. rep. National Institute of Standards and Technology, Oct. 2020. doi: 10.6028/NIST.SP.800-208.
- [11] David McGrew, Michael Curcio, and Scott Fluhrer. *Leighton-Micali Hash-Based Signatures*. en. Tech. rep. RFC Editor, Apr. 2019, RFC8554. doi: 10.17487/RFC8554.
- [12] Andreas Hülsing, Stefan-Lukas Gazdag, Denis Butin, and Johannes A. Buchmann. "Hash-based Signatures : An Outline for a New Standard". In: 2014.
- [13] National Institute of Standards and Technology. *Stateless Hash-Based Digital Signature Standard*. Tech. rep. NIST FIPS 205 ipd. Gaithersburg, MD: National Institute of Standards and Technology, Aug. 2023, NIST FIPS 205 ipd. doi: 10.6028/NIST.FIPS.205.ipd.
- [14] Leslie Lamport. *Constructing digital signatures from a one-way function*. Tech. rep. SRI-CSL-98, SRI International Computer Science Laboratory, 1978.
- [15] Andreas Hülsing. "W-OTS+ – Shorter Signatures for Hash-Based Signature Schemes". In: *Progress in Cryptology – AFRICACRYPT 2013*. Ed. by Amr Youssef, Abderrahmane Nitaj, and Aboul Ella Hassanien. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 173–188. ISBN: 978-3-642-38553-7.
- [16] Leonid Reyzin and Natan Reyzin. "Better than BiBa: Short One-Time Signatures with Fast Signing and Verifying". In: *Information Security and Privacy*. Ed. by Lynn Batten and Jennifer Seberry. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 144–153. ISBN: 978-3-540-45450-2.
- [17] Daniel J. Bernstein et al. "The SPHINCS+ Signature Framework". In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. CCS '19. London, United Kingdom: Association for Computing Machinery, 2019, pp. 2129–2146. ISBN: 9781450367479. doi: 10.1145/3319535.3363229.
- [18] Ralph C. Merkle and Martin E. Hellman. "Hiding information and signatures in trapdoor knapsacks". In: *IEEE Transactions on Information Theory* 24.5 (1978), pp. 525–530.
- [19] Arjen K. Lenstra, Hendrik Willem Lenstra, and László Lovász. "Factoring polynomials with rational coefficients". In: *Mathematische Annalen* 261.4 (1982), pp. 515–534.

- [20] Miklós Ajtai. "The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract)". en. In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing – STOC '98*. Dallas, Texas, United States: ACM Press, 1998, pp. 10–19. ISBN: 978-0-89791-962-3. doi: 10.1145/276698.276705.
- [21] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*. Baltimore, MD, USA, 2005, pp. 84–93.
- [22] Robert J. McEliece. "A Public-Key Cryptosystem Based On Algebraic Coding Theory". In: *Deep Space Network Progress Report 44* (Jan. 1978). ADS Bibcode: 1978DSNPR..44..114M, pp. 114–116. url: <https://ui.adsabs.harvard.edu/abs/1978DSNPR..44..114M>.
- [23] Raphael Overbeck and Nicolas Sendrier. "Code-based cryptography". In: *Post-Quantum Cryptography*. Ed. by Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 95–145. ISBN: 978-3-540-88702-7. doi: 10.1007/978-3-540-88702-7_4.
- [24] Elwyn Berlekamp, Robert McEliece, and Henk Van Tilborg. "On the inherent intractability of certain coding problems (Corresp.)" en. In: *IEEE Transactions on Information Theory* 24.3 (May 1978), pp. 384–386. ISSN: 0018-9448. doi: 10.1109/TIT.1978.1055873.
- [25] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. "Attacking and Defending the McEliece Cryptosystem". In: *Post-Quantum Cryptography*. Ed. by Johannes Buchmann and Jintai Ding. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 31–46. ISBN: 978-3-540-88403-3.
- [26] Henk C. A. Van Tilborg and Sushil Jajodia, eds. *Encyclopedia of Cryptography and Security*. en. Boston, MA: Springer US, 2011. ISBN: 978-1-4419-5905-8 978-1-4419-5906-5. doi: 10.1007/978-1-4419-5906-5.
- [27] Belkacem Imine, Naima Hadj-Said, and Adda Ali-Pacha. "McEliece cryptosystem based on Plotkin construction with QC-MDPC and QC-LDPC codes". In: (2022). Publisher: [object Object] Version Number: 3. doi: 10.48550/ARXIV.2211.14206.
- [28] Martin R. Albrecht et al. *BIKE: Bit Flipping Key Encapsulation*. Submission to NIST's post-quantum cryptography standardization process. Round 3 Submission. 2020. url: <https://bikesuite.org/files/BIKE-Spec-R3.pdf>.
- [29] Christophe Aguilar Melchor et al. *HQC: Hamming Quasi-Cyclic*. Submission to NIST's post-quantum cryptography standardization process. Round 3 Submission. 2020. url: https://pqc-hqc.org/doc/hqc-specification_2021-04-21.pdf.
- [30] Denis X. Charles, Kristin E. Lauter, and Eyal Z. Goren. "Cryptographic Hash Functions from Expander Graphs". en. In: *Journal of Cryptology* 22.1 (Jan. 2009), pp. 93–113. ISSN: 0933-2790, 1432-1378. doi: 10.1007/s00145-007-9002-x.
- [31] Jean-Marc Couveignes. *Hard Homogeneous Spaces*. Cryptology ePrint Archive, Paper 2006/291. 2006. url: <https://eprint.iacr.org/2006/291>.
- [32] Alexander Rostovtsev and Anton Stolbunov. "Public-Key Cryptosystem Based on Isogenies". In: *IACR Cryptol. ePrint Arch.* 2006 (2006), p. 145. url: <https://api.semanticscholar.org/CorpusID:30785494>.
- [33] Steven D. Galbraith, Florian Hess, and Nigel P. Smart. "Extending the GHS Weil Descent Attack". In: *Advances in Cryptology — EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 29–44. ISBN: 978-3-540-46035-0.

- [34] David Jao, Stephen D. Miller, and Ramarathnam Venkatesan. “Do All Elliptic Curves of the Same Order Have the Same Difficulty of Discrete Log?” In: *International Conference on the Theory and Application of Cryptology and Information Security*. 2004. URL: <https://api.semanticscholar.org/CorpusID:9710611>.
- [35] Denis Xavier Charles, Eyal Z. Goren, and Kristin E. Lauter. “Cryptographic Hash Functions from Expander Graphs”. In: *J. Cryptology* 22 (2009), pp. 93–113. doi: 10.1007/s00145-007-9002-x.
- [36] Andrew Childs, David Jao, and Vladimir Soukharev. “Constructing elliptic curve isogenies in quantum subexponential time”. en. In: *Journal of Mathematical Cryptology* 8.1 (Feb. 2014), pp. 1–29. ISSN: 1862-2984, 1862-2976. doi: 10.1515/jmc-2012-0016.
- [37] Greg Kuperberg. “A Subexponential-Time Quantum Algorithm for the Dihedral Hidden Subgroup Problem”. In: *SIAM Journal on Computing* 35.1 (2005), pp. 170–188. doi: 10.1137/S00975-39703436345.
- [38] David Jao and Luca De Feo. “Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies”. In: *Post-Quantum Cryptography*. Ed. by Bo-Yin Yang. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 19–34. ISBN: 978-3-642-25405-5.
- [39] Jintai Ding and Dieter Schmidt. “Rainbow, a New Multivariable Polynomial Signature Scheme”. en. In: *Applied Cryptography and Network Security*. Ed. by David Hutchison et al. Vol. 3531. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 164–175. ISBN: 978-3-540-31542-1. doi: 10.1007/11496137_12.
- [40] European Union Agency for Cybersecurity. *Post-quantum cryptography: current state and quantum mitigation*. eng. LU: Publications Office, 2021. URL: <https://data.europa.eu/doi/10.2824/92307>.
- [41] P. Zimmermann, A. Johnston, and J. Callas. *Z RTP: Media Path Key Agreement for Unicast Secure RTP*. en. Tech. rep. RFC6189. RFC Editor, Apr. 2011, RFC6189. doi: 10.17487/rfc6189.
- [42] Jason A. Donenfeld. “WireGuard: Next Generation Kernel Network Tunnel”. In: NDSS. 2017.
- [43] Trevor Perrin. *The Noise Protocol Framework*. NoiseProtocol.org, Revision 34, Official/Unstable. 2018. URL: <https://noiseprotocol.org/noise.pdf>.
- [44] Jacob R. Appelbaum, Chloe R. Martindale, and Sinli Peter Wu. “Tiny wireguard tweak”. English. In: *Progress in Cryptology – AFRICACRYPT 2019 - 11th International Conference on Cryptology in Africa, Proceedings*. Ed. by Abderrahmane Nitaj, Tajjeeddine Rachidi, and Johannes Buchmann. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 11th International Conference on the Theory and Applications of Cryptographic Techniques in Africa, Africacrypt 2019. Germany: Springer, Jan. 2019, pp. 3–20. ISBN: 978-3-030-23695-3. doi: 10.1007/978-3-030-23696-0_1.
- [45] Andreas Hulsing, Kai-Chun Ning, Peter Schwabe, Florian Weber, and Philip R. Zimmermann. “Post-quantum WireGuard”. In: *2021 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2021, pp. 304–321. ISBN: 978-1-72818-934-5. doi: 10.1109/SP40001.2021.00030.

Μέρος V

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ Α

ΟΔΗΓΙΕΣ ΧΡΗΣΗΣ ΤΟΥ CRYPTOTool 2

A.1 CrypTool 2

Το CrypTool 2 είναι ένα γραφικό περιβάλλον ανοικτού λογισμικού το οποίο έχει αναπτυχθεί με στόχο την εξοικείωση και εκμάθηση βασικών εννοιών της κρυπτογραφίας. Παρέχεται δωρεάν στη σελίδα <https://www.cryptool.org/ct2/>. Είναι πολύ απλό στη χρήση του και ευρύτατα διαδεδομένο λογισμικό ηλεκτρονικής μάθησης στον τομέα της κρυπτογραφίας. Μέσα σε αυτό ένας μεγάλος αριθμός εργαλείων ανάλυσης και αλγόριθμοι έχουν υλοποιηθεί αποτελεσματικά. Η γραφική διεπαφή και η πλούσια ηλεκτρονική του τεκμηρίωση δίνει στο χρήστη τη δυνατότητα, να γνωρίσει τη κρυπτογραφία, δημιουργώντας δικές του διατάξεις, δοκιμάζοντας αλγορίθμους και μελετώντας τα αποτελέσματα.

A.1.1 Περιβάλλον Εργασίας

Το Σχήμα A.1 δείχνει την αρχική εικόνα του προγράμματος όταν το CrypTool 2 ξεκινά. Το Startcenter είναι η πρώτη εικόνα που εμφανίζεται και από εδώ ο χρήστης μπορεί να μεταφερθεί σε οποιαδήποτε άλλη λειτουργία επιλέγοντάς την.

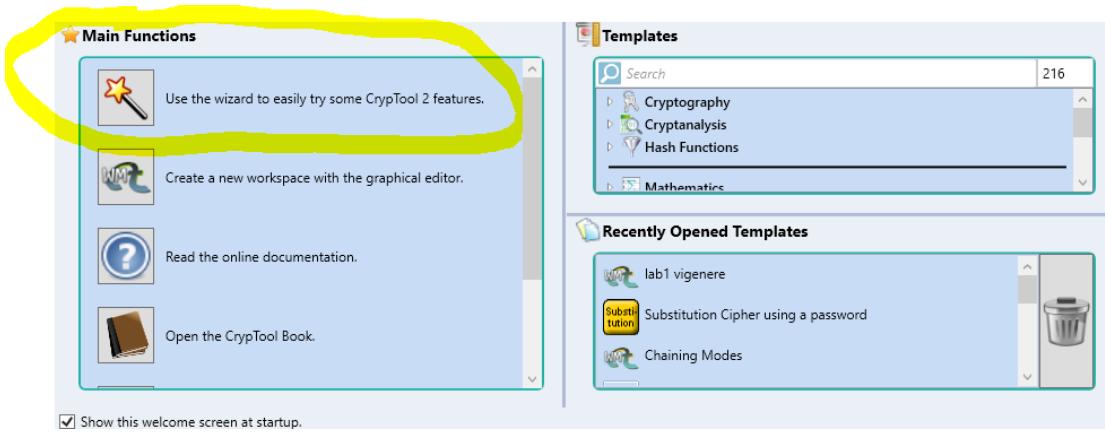
A.1.2 Το εργαλείο Wizard

Ο απλούστερος τρόπος χρήσης και εξοικείωσης με το CrypTool 2 είναι από το εργαλείο Wizard: Startcenter → Use the wizard (βλέπε Σχήμα A.1, πρώτη επιλογή). Ο χρήστης συμπληρώνει βήμα-προς-βήμα επιλέγοντας τις βασικές λειτουργίες από μια διαθέσιμη λίστα επιλογών (βλέπε Σχήμα A.2)

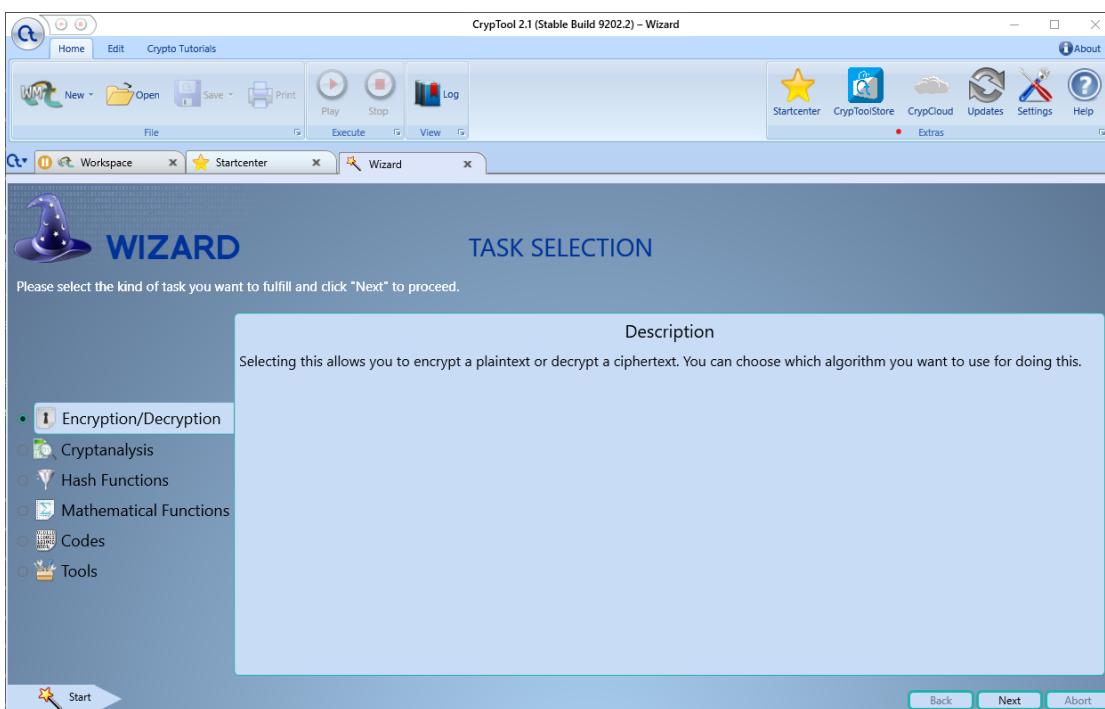
A.1.3 To Workspace

Το Workspace αποτελεί το βασικό χώρο εργασίας του CrypTool 2, καθώς επιτρέπει στον χρήστη να δημιουργεί διάφορα σενάρια κρυπτογράφησης (βλέπε Σχήμα A.3).

Ο οριζόντιος χώρος στην επάνω πλευρά είναι οι επιλογές μενού στις βασικές λειτουργίες (αποθήκευση,



Σχήμα A.1: Η πρώτη εικόνα του προγράμματος κατά την έναρξη του CrypTool 2.

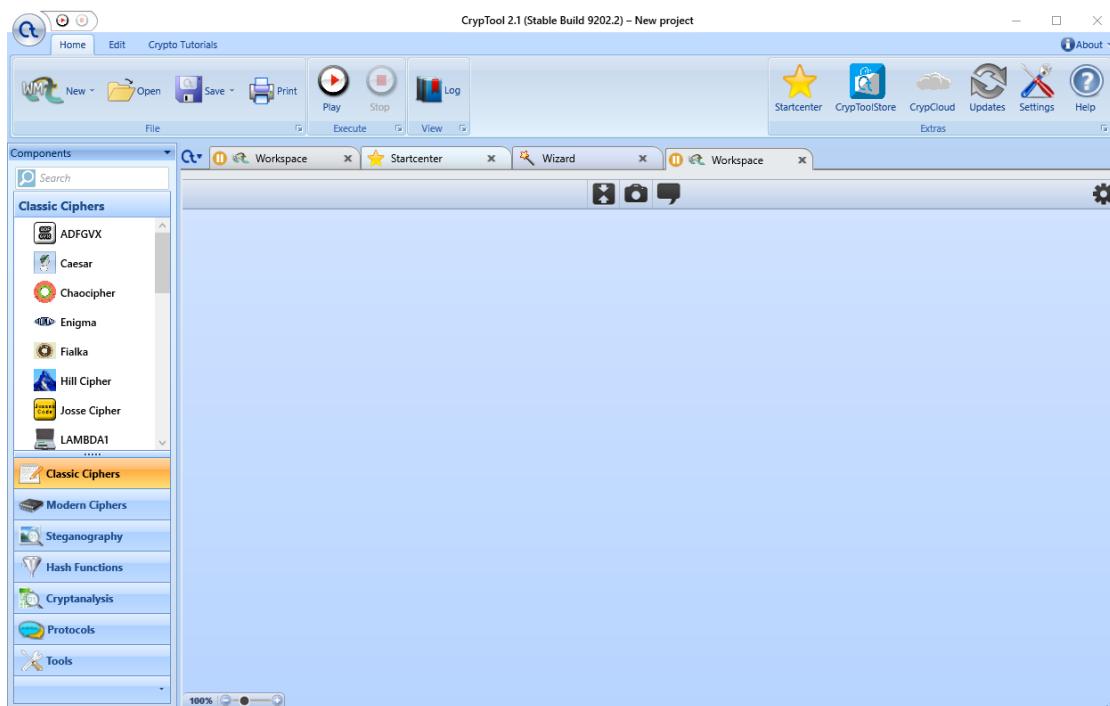


Σχήμα A.2: Επιλέγοντας αλγόριθμο κρυπτογράφησης στο CrypTool 2 με το εργαλείο Wizard.

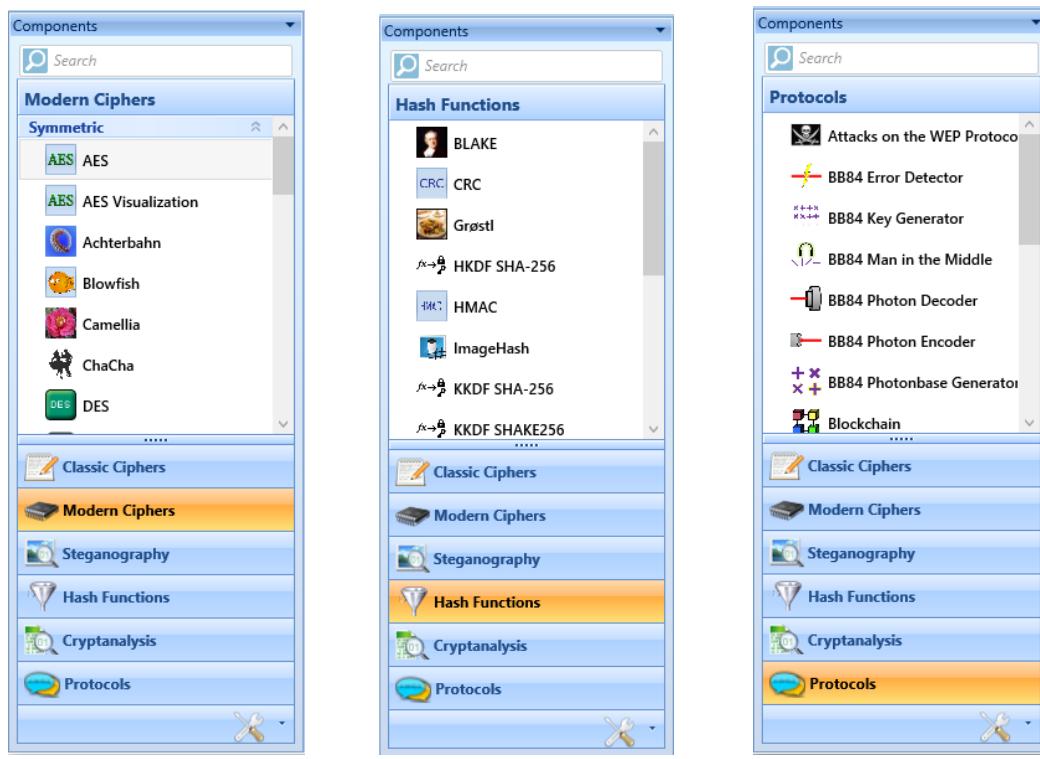
επιλογή αρχείων, ενημερώσεις, ρυθμίσεις και εκτέλεση του σεναρίου). Η κάθετη αριστερή πλευρά είναι η εργαλειοθήκη των επιλογών, ενώ ο υπόλοιπος χώρος είναι η περιοχή εργασίας για την υλοποίηση της σύνδεσης των απαραίτητων μερών.

Για τη δημιουργία ενός σεναρίου, ο χρήστης ακολουθεί τα παρακάτω βήματα:

1. Επιλογή και τοποθέτηση των στοιχείων (components) που απαιτούνται για να εκτελεστεί το σενάριο, από μια λίστα διαθέσιμων επιλογών (drag and drop) (βλέπε Σχήμα A.4). Επιπλέον, είναι εύκολη η αναζήτηση πληκτρολογώντας τα αρχικά γράμματα του αλγόριθμου.
2. Δημιουργία των κατάλληλων συνδέσεων μεταξύ των στοιχείων αυτών.
3. Εκτέλεση (Play) και εμφάνιση των αποτελεσμάτων της εκτέλεσης. Τα αποτελέσματα που λαμβάνονται στη συνέχεια μελετώνται και συγκρίνονται ή χρησιμοποιούνται σε νέα κρυπτογράφηση ή αποκρυπτογράφηση.



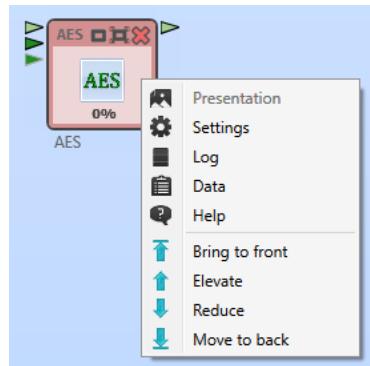
Σχήμα A.3: Το περιβάλλον εργασίας του CrypTool 2.



Σχήμα A.4: Εργαλειοθήκη διαθέσιμων επιλογών του CrypTool 2.

Δύο σημαντικές βοηθητικές λειτουργίες του CrypTool 2 είναι το δεξί "κλικ" πάνω σε αυτό. Σε αυτήν την περίπτωση το πρόγραμμα δίνει περισσότερες πληροφορίες σχετικά με τον τρόπο χρήσης και τις λειτουργίες του (βλέπε Σχήμα A.5). Επιπλέον, πατώντας το πλήκτρο F1 ενώ έχει επιλεχθεί κάποιο στοιχείο, εμφανίζεται

η σχετική βοήθεια (online help) για το στοιχείο αυτό.

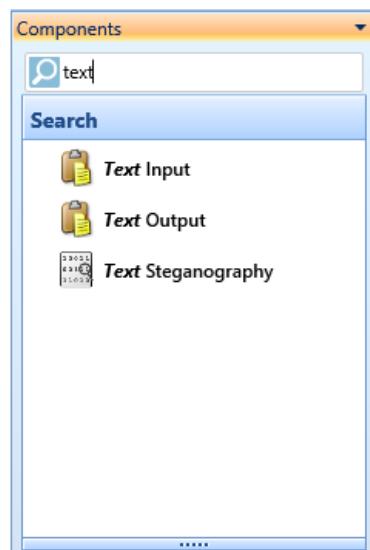


Σχήμα A.5: Μενού διαθέσιμων επιλογών ενός στοιχείου (επιλογή δεξί κλικ πάνω στο στοιχείο).

A.1.4 Παράδειγμα Δημιουργίας Σεναρίου με το CrypTool 2

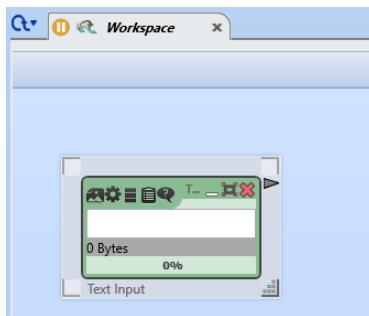
Στο παρακάτω παράδειγμα θα δούμε πως μπορούμε να υλοποιήσουμε ένα σενάριο χρήσης της συνάρτησης σύνοψης (hash function) SHA-2. Όπως κάθε συνάρτηση σύνοψης, έτσι και η συγκεκριμένη λαμβάνει ως τιμή εισόδου ένα string τυχαίου μήκους και επιστρέφει μία “σύνοψη” (digest). Για το σενάριο αυτό θα χρειαστούμε 3 στοιχεία (components): ένα στοιχείο για το κείμενο εισόδου, ένα στοιχείο για τη συνάρτηση σύνοψης και ένα στοιχεία για την έξοδο.

- Για την εύρεση του στοιχείου “Text Input” το αναζητούμε στο πεδίο Search που βρίσκεται στην εργαλειοθήκη Components ώστε να εμφανιστεί η αντίστοιχη επιλογή (βλέπε Σχήμα A.6).

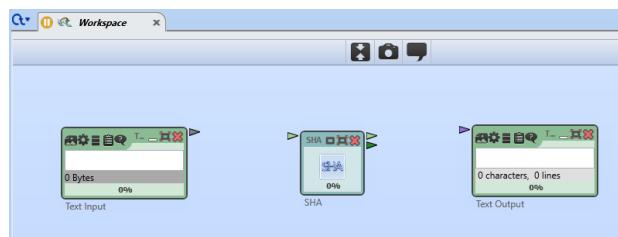


Σχήμα A.6: Εύρεση στοιχείων σχετικών με τη λέξη “text”.

- Χρησιμοποιώντας την τεχνική “drag and drop” εισάγουμε το επιλεγμένο στοιχείο “Text Input” στο περιβάλλον εργασίας (βλέπε Σχήμα A.7).
- Με τον ίδιο τρόπο εισάγουμε μέσα στο ίδιο Workspace τα στοιχεία “SHA” και “Text Output” (βλέπε Σχήμα A.8).

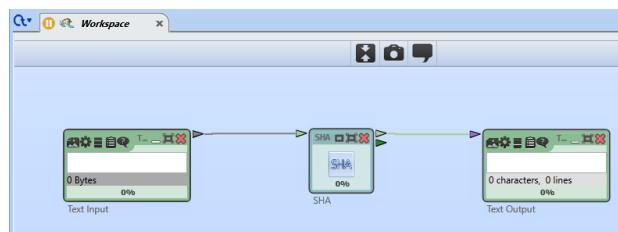


Σχήμα A.7: Εισαγωγή του στοιχείου “Text Input” στο περιβάλλον εργασίας.



Σχήμα A.8: Εισαγωγή των στοιχείων “SHA” και “Text Output”.

- Ενώνουμε τα στοιχεία με τον τρόπο που φαίνεται στο Σχήμα A.9 έτσι ώστε το “Text input” να αποτελέσει είσοδο στη συνάρτηση “SHA” ενώ η έξοδος της συνάρτησης να αποθηκευτεί στο “Text output”. Προσέξτε ότι όλα τα «βέλη» δεν είναι ίδια. Ακουμπήστε το δείκτη του ποντικιού πάνω από κάθε βέλος ώστε να διαπιστώσετε τις διαφορετικές λειτουργικότητες του κάθε βέλους.



Σχήμα A.9: Διασύνδεση όλων των στοιχείων.

- Από τα “Settings” της συνάρτησης SHA μπορείτε να επιλέξετε μεταξύ των συναρτήσεων SHA-1, SHA-256, SHA-384 και SHA-512.
- Εισάγετε στο εργαλείο “Text Input” το κείμενο, τη σύνοψη του οποίου θέλετε να υπολογίσετε, και επιλέξτε το “Play” στην εργαλειοθήκη “Execute”. Θα παρατηρήσετε την “σύνοψη” που δημιουργείται από τη συνάρτηση στο εργαλείο “Text Output”. Όσο αλλάζετε το κείμενο εισόδου, η συνάρτηση δημιουργεί δυναμικά τη νέα σύνοψη. Για να διακόψετε την εκτέλεση του σεναρίου, επιλέξτε το “Stop” στην εργαλειοθήκη “Execute”.

ΠΑΡΑΡΤΗΜΑ Β

ΑΠΑΝΤΗΣΕΙΣ ΕΡΩΤΗΣΕΩΝ - ΛΥΣΕΙΣ ΑΣΚΗΣΕΩΝ

Β.1 Κεφάλαιο 1

Β.1.1 Λύσεις Εργασιών

1.6.1 (σελ. 25): Ενδεικτική λύση για το κύριο σκέλος της εργασίας είναι διαθέσιμη [εδώ](#). Ενώ ενδεικτικές λύσεις για τα επιμέρους υποερωτήματα (1) και (2) είναι διαθέσιμες [εδώ](#) και [εδώ](#), αντίστοιχα.

Κύριο σκέλος:



Υποερώτημα (1):



Υποερώτημα (2):



1.6.2 (σελ. 25): Ενδεικτική λύση της εργασίας είναι διαθέσιμη [εδώ](#).



Β.2 Κεφάλαιο 2

Β.2.1 Λύσεις Ασκήσεων

2.7.1 (σελ. 45): Οι απαντήσεις είναι οι εξής:

- (1) Το $n = p \cdot q = 55$ και το $\Phi(n) = (p-1) \cdot (q-1) = 4 \cdot 10 = 40$.
- (2) Το $e = 3$ είναι μια σωστή επιλογή γιατί $\gcd(3, 40) = 1$ (σημειώστε ότι το 3 είναι και πρώτος αριθμός). Δεδομένου ότι $3 \cdot d \equiv 1 \pmod{40}$, τότε ο ιδιωτικός εκθέτης d της Αλίκης θα είναι $d = 27$ από την στιγμή που $3 \cdot 27 = 81 \equiv 1 \pmod{40}$.
- (3) Το κρυπτοκείμενο c θα είναι $c = 4^3 \pmod{55} = 64 \pmod{55} = 9$.
- (4) Η Αλίκη λαμβάνει το c και υπολογίζει $9^{27} \pmod{55} = 4$ που είναι ίσο με το μήνυμα m .

2.7.2 (σελ. 45): Η υπογραφή είναι έγκυρη εάν $m \equiv s^e \pmod{n}$. Οπότε στην περίπτωση μας ο υπολογισμός που πρέπει να γίνει είναι ο εξής: $182^{13} \pmod{221} = 65$, το οποίο είναι έγκυρο.

2.7.3 (σελ. 45): Οι απαντήσεις είναι οι εξής:

- (1) Δεδομένου ότι $493 = p \cdot q$ και $448 = (p-1)(q-1)$, τότε αντικαθιστούμε στην δεύτερη εξίσωση το $q = \frac{493}{p}$ και λύνουμε την εξίσωση ως εξής:

$$\begin{aligned} 448 &= (p-1)(q-1) \Leftrightarrow 448 = (p-1)\left(\frac{493}{p} - 1\right) \\ &\Leftrightarrow 448 = (p-1)\frac{493-p}{p} \\ &\Leftrightarrow 448 = \frac{493p - p^2 - 493 + p}{p} \\ &\Leftrightarrow 448p = -p^2 + 494p - 493 \\ &\Leftrightarrow p^2 - 46p + 493 = 0 \end{aligned}$$

Υπολογίζουμε την διακρίνουσα $\Delta = (-46)^2 - 4 \cdot 1 \cdot 493 = 144$, οπότε οι λύσεις είναι:

$$p = \frac{46 \pm \sqrt{144}}{2 \cdot 1} = 29 \text{ ή } 17, \text{ και το } q = \frac{493}{p} = 17 \text{ ή } 29$$

- (2) Δεδομένου ότι $d \equiv e^{-1} \pmod{\Phi(n)}$, τότε το d μπορεί να υπολογιστεί εύκολα ως εξής:

$$d = 3^{-1} \pmod{448} = 229$$

2.7.4 (σελ. 45): Το αποκρυπτογραφημένο μήνυμα m δίνεται από την εξίσωση:

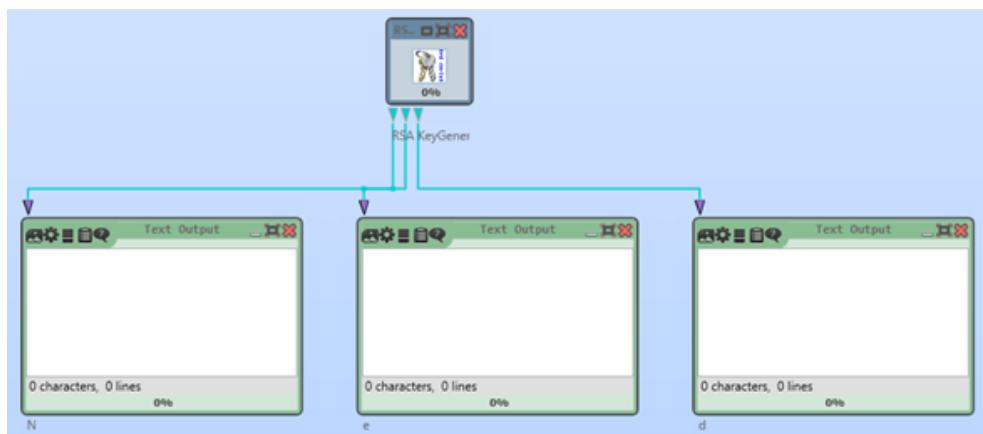
$$\begin{aligned} m &= \gamma^{-a} \cdot \delta \pmod{p} = \gamma^{p-a-1} \cdot \delta \pmod{p} \\ &= 78^{283-7-1} \cdot 218 \pmod{283} \\ &= 78^{275} \cdot 218 \pmod{283} \\ &= 116 \cdot 218 \pmod{283} \\ &= 101 \end{aligned}$$

2.7.5 (σελ. 45): Το κρυπτογραφημένο μήνυμα c δίνεται από την εξίσωση:

$$\begin{aligned} c &= g^m \cdot r^n \pmod{n^2} = 4886^{123} \cdot 48^{221} \pmod{221^2} \\ &= 42021 \cdot 44086 \pmod{48841} \\ &= 47517 \end{aligned}$$

B.2.2 Λύσεις Εργασιών

2.7.1 (σελ. 45): Για τη δημιουργία των ασύμμετρων κλειδιών του κρυπτοσυστήματος RSA, να χρησιμοποιήσετε το component του CrypTool 2 με όνομα *RSA Key Generator* (Σχήμα B.1). Η γεννήτρια αυτή παράγει το δημόσιο (e, n) και το ιδιωτικό κλειδί του ζεύγους (d, n). Για την ολοκλήρωση της εργασίας θα πρέπει να χρησιμοποιήσετε τις κατάλληλες εξόδους του *RSA Key Generator* τόσο για κρυπτογράφηση όσο και για αποκρυπτογράφηση. Ένα ενδεικτικό CrypTool 2 Project που υλοποιεί αυτήν την εργασία είναι διαθέσιμο [εδώ](#).



Σχήμα B.1: Γεννήτρια κλειδιών RSA στο περιβάλλον CrypTool 2.

2.7.2 (σελ. 47): Κάποιες ενδεικτικές εξηγήσεις στα ερωτήματα αυτής της εργασίας είναι τα εξής:

- (1) Το αποτέλεσμα αποκρυπτογράφησης δεν είναι -1 γιατί έχουν γίνει πράξεις με modulo n και με το πεδίο τιμών να είναι θετικοί ακέραιοι αριθμοί (Z_n). Επιπλέον, ο padded-RSA παράγει κάθε φορά διαφορετικό κρυπτοκείμενο για το ίδιο μήνυμα εισόδου γιατί προσθέτει ψηφία τυχαιοποίησης στο αρχικό μήνυμα.
- (2) Στην περίπτωση που δεν γίνεται χρήση προϋπολογισμένων τιμών r και g , ο αλγόριθμος επιλογής τιμών για τα r και g είναι αυτός που καθυστερεί για ελεγχθεί διεξοδικά εάν πληρούνται τα κριτήρια επιλογής τους.
- (3) Το αποτέλεσμα αποκρυπτογράφησης του γινομένου των δύο κρυπτοκειμένων είναι το άθροισμα τους, κάτι το οποίο είναι γνωστό ως αθροιστική ομομορφική ιδιότητα του κρυπτοσυστήματος Paillier και παρουσιάζεται αναλυτικά στο Κεφάλαιο 8.
- (4) Από την στιγμή που το αρχικό μήνυμα άλλαξε είναι προφανές ότι δεν ισχύει πλέον η υπογραφή που είχε παραχθεί.
- (5) Η υπογραφή μπορεί να είναι έγκυρη ακόμη και για μηνύματα που δεν είναι ακέραιοι αριθμοί, απλά σε αυτή την περίπτωση έχει μεγάλη σημασία η κωδικοποίηση που θα χρησιμοποιηθεί. Από την στιγμή που το κείμενο μετατράπηκε σε *BigInteger*, το αποτέλεσμα της εκτύπωσης θα είναι απλά ένας ακέραιος δεκαδικός αριθμός. Όλες οι πράξεις στα κρυπτοσυστήματα δημοσίου κλειδιού γίνονται στο δεκαδικό σύστημα (εκτός κάποιων πιθανών εξαιρέσεων).

B.3 Κεφάλαιο 3

B.3.1 Λύσεις Ασκήσεων

3.7.1 (σελ. 70): Χρησιμοποιώντας τη δεδομένη συνάρτηση σύνοψης $h(x) = x \bmod 10$ υπολογίζουμε τα εξής:

- $h(9679) = 9679 \bmod 10 = 9$
- $h(1989) = 1989 \bmod 10 = 9$
- $h(4199) = 4199 \bmod 10 = 9$
- $h(1471) = 1471 \bmod 10 = 1$
- $h(6171) = 6171 \bmod 10 = 1$

Όπως μπορείτε να δείτε, οι τιμές 9679, 1989 και 4199 έχουν ως τιμή σύνοψης το 9, ενώ οι τιμές 1471 και 6171 έχουν ως τιμή σύνοψης το 1. Επομένως, οι προτάσεις (i) και (ii) είναι αληθείς, ενώ οι (iii) και (iv) είναι ψευδείς.

3.7.2 (σελ. 70): Αυτή η συνάρτηση σαφώς ικανοποιεί το (α), ωστόσο, τα (β) και (γ) αποτυγχάνουν. Δεδομένου μιας σύνοψης μηνύματος y , έστω ότι $m = y$, τότε $h(m) = y$ και επομένως η συνάρτηση σύνοψης h δεν είναι μονόδρομος. Όμοια, εάν επιλέξετε οποιεσδήποτε δύο τιμές m_1 και m_2 που συγκλίνουν με $\bmod n$, τότε:

$$h(m_1) = h(m_2)$$

και επομένως η συνάρτηση σύνοψης h δεν είναι ανθεκτική σε συγκρούσεις.

3.7.3 (σελ. 71): Οι απαντήσεις είναι οι εξής:

(1) Για το πρωτόκολλο A', τα βήματα που κάνει ο παραλήπτης είναι τα εξής:

- Αποκρυπτογραφεί χρησιμοποιώντας το συμμετρικό κλειδί k_1 και λαμβάνει το $x \parallel H(k_2 \parallel x)$. Εξάγει από αυτό το μήνυμα x .
- Υπολογίζει την σύνοψη του $k_2 \parallel x$ και συγκρίνει το αποτέλεσμα με το $H(k_2 \parallel x)$.

Ενώ για το πρωτόκολλο B' πραγματοποιεί τα εξής:

- Εξάγει από το y το μήνυμα x . Αποκρυπτογραφεί χρησιμοποιώντας το ιδιωτικό κλειδί s_k και λαμβάνει το $H(x)$.
- Υπολογίζει την σύνοψη του x και συγκρίνει το αποτέλεσμα με το $H(x)$.

(2) Στην περίπτωση του πρωτοκόλλου A', ισχύουν τα εξής:

- *Εμπιστευτικότητα* – Παρέχεται μέσω της κρυπτογράφησης.
- *Ακεραιότητα* – Παρέχεται μέσω της συνάρτησης σύνοψης.
- *Μη-αποποίηση* – Δεν παρέχεται γιατί και οι δύο συμμετέχοντες μπορούν να ισχυριστούν ότι ο άλλος έχει δημιουργήσει το μήνυμα.

Ενώ στην περίπτωση του πρωτοκόλλου B', ισχύουν τα εξής:

- *Εμπιστευτικότητα* – Δεν παρέχεται γιατί το μήνυμα x δεν είναι κρυπτογραφημένο.
- *Ακεραιότητα* – Δεν παρέχεται γιατί ο οποιοσδήποτε μπορεί να αντικαταστήσει το μήνυμα x και να υπολογίσει την σύνοψη $H(x)$.
- *Μη-αποποίηση* – Δεν παρέχεται γιατί ο οποιοσδήποτε μπορεί να δημιουργήσει ένα τέτοιο μήνυμα y κάνοντας χρήση του κλειδιού p_k που είναι δημόσια διαθέσιμη.

B.3.2 Λύσεις Εργασιών

3.7.1 (σελ. 71): Για τη δημιουργία των συναρτήσεων σύνοψης MD5, Whirlpool, SHA-1, SHA-256, SHA3-256, και BLAKE-512, να χρησιμοποιήσετε τα components του CrypTool 2 με τα εξής ονόματα *MD5*, *Whirlpool*, *SHA*, *SHA*, *Keccak*, και *BLAKE*, αντίστοιχα, κάνοντας τις απαραίτητες ρυθμίσεις (όπου απαιτείται) σε κάθε component. Για την ολοκλήρωση της εργασίας θα πρέπει να οπτικοποιήσετε τις εξόδους των διαφόρων συναρτήσεων σύνοψης σε έξι ξεχωριστά components *Text Output*. Ένα ενδεικτικό CrypTool 2 Project που υλοποιεί αυτήν την εργασία είναι διαθέσιμο [εδώ](#).



3.7.2 (σελ. 72): Κάποιες ενδεικτικές απαντήσεις στα ερωτήματα αυτής της εργασίας είναι τα εξής:

- (1) Το αποτέλεσμα σύνοψης του MD5 είναι τελείως διαφορετικό με αυτή και μόνο την μικρή αλλαγή:
 - “Cryptography” → “64ef07ce3e4b420c334227eecb3b3f4c”
 - “cryptography” → “e0d00b9f337d357c6faa2f8ceae4a60d”
- (2) Η χρήση του salt “hello” αλλάζει πλήρως την σύνοψη εξόδου του SHA-1 και ενδείκνυται για την αποθήκευση κωδικών πρόσβασης και στις ψηφιακές υπογραφές. Ακολουθούν οι συνόψεις που παράχθηκαν:
 - SHA-1 του “Cryptography” χωρίς salt → “b804ec5a0d83d19d8db908572f51196505d09f98”
 - SHA-1 του “Cryptography” με salt → “f0f4262847cf8a76d53a8cc8bbb23a1d7567561a”
- (3) Όποια αλλαγή και να γίνει στο αρχικό μήνυμα, σε περιεχόμενο και μήκος, έχει ως αποτέλεσμα συνόψεις του ίδιου σταθερού μήκους και τελείως διαφορετικού περιεχομένου δεδομένων εξόδου.

B.4 Κεφάλαιο 4

B.4.1 Λύσεις Ασκήσεων

4.8.1 (σελ. 93): Χρησιμοποιώντας τη συνάρτηση του αλγορίθμου Blum-Blum-Shub $x_{n+1} = x_n^2 \text{ mod } M$, όπου $M = p \cdot q = 11 \cdot 23 = 253$, υπολογίζουμε τους πρώτους 6 αριθμούς που παράγονται:

- $x_0 = x_{-1}^2 \text{ mod } 253 = 3^2 \text{ mod } 253 = 9$
- $x_1 = x_0^2 \text{ mod } 253 = 9^2 \text{ mod } 253 = 81$
- $x_2 = x_1^2 \text{ mod } 253 = 81^2 \text{ mod } 253 = 236$
- $x_3 = x_2^2 \text{ mod } 253 = 236^2 \text{ mod } 253 = 36$
- $x_4 = x_3^2 \text{ mod } 253 = 36^2 \text{ mod } 253 = 31$
- $x_5 = x_4^2 \text{ mod } 253 = 31^2 \text{ mod } 253 = 202$

Με βάση αυτούς βρίσκουμε τα εξής:

(1) Κάνοντας χρήση της τεχνικής του bit ισοτιμίας, υπολογίζουμε τα εξής:

- $9_{dec} = 1001_{bin} = 1 + 0 + 0 + 1 \pmod{2} = 0$
- $81_{dec} = 01010001_{bin} = 0 + 1 + 0 + 1 + 0 + 0 + 0 + 1 \pmod{2} = 1$
- $236_{dec} = 11101100_{bin} = 1 + 1 + 1 + 0 + 1 + 1 + 0 + 0 \pmod{2} = 1$
- $36_{dec} = 00100100_{bin} = 0 + 0 + 1 + 0 + 0 + 1 + 0 + 0 \pmod{2} = 0$
- $31_{dec} = 00011111_{bin} = 0 + 0 + 0 + 1 + 1 + 1 + 1 + 1 \pmod{2} = 1$
- $202_{dec} = 11001010_{bin} = 1 + 1 + 0 + 0 + 1 + 0 + 1 + 0 \pmod{2} = 0$

(2) Κάνοντας χρήση της τεχνικής του λιγότερου σημαντικού bit, βρίσκουμε τα εξής:

- $9_{dec} = 1001_{bin} = 1$
- $81_{dec} = 01010001_{bin} = 1$
- $236_{dec} = 11101100_{bin} = 0$
- $36_{dec} = 00100100_{bin} = 0$
- $31_{dec} = 00011111_{bin} = 1$
- $202_{dec} = 11001010_{bin} = 0$

4.8.2 (σελ. 93): Χρησιμοποιώντας τη συνάρτηση του αλγορίθμου Blum-Micali $x_{i+1} = g^{x_i} \pmod{p} = 6^{x_i} \pmod{17}$, υπολογίζουμε τους πρώτους 4 αριθμούς που παράγονται:

- $x_1 = g^{x_0} \pmod{p} = 6^9 \pmod{17} = 11$
- $x_2 = g^{x_1} \pmod{p} = 6^{11} \pmod{17} = 5$
- $x_3 = g^{x_2} \pmod{p} = 6^5 \pmod{17} = 7$
- $x_4 = g^{x_3} \pmod{p} = 6^7 \pmod{17} = 14$

Με βάση αυτούς βρίσκουμε τα εξής:

- $11 > (p-1)/2 = 8 \rightarrow 0$
- $5 \leq (p-1)/2 = 8 \rightarrow 1$
- $7 \leq (p-1)/2 = 8 \rightarrow 1$
- $14 > (p-1)/2 = 8 \rightarrow 0$

B.4.2 Λύσεις Εργασιών

4.8.1 (σελ. 93): Οι απαντήσεις στα ερωτήματα αυτής της εργασίας είναι τα εξής:

- (1) Χρειάζεται απλά να αλλάξετε το μέγεθος του πίνακα data για να παράγεται περισσότερα τυχαία bytes, δηλ., “byte[] data = new byte[64];”.
- (2) Για να αυξήσετε την παρεχόμενη ασφάλεια στον μηχανισμό HASH-DRBG, χρειάζεται απλά στην συνάρτηση buildHASH_DRBG να κάνετε την εξής αλλαγή “.setSecurityStrength(512)”. Ο λόγος για τον οποίο χτυπάει σφάλμα κατά την εκτέλεση οφείλεται στο γεγονός ότι η υψηλότερη παρεχόμενη ασφάλεια για τον SHA-512, σύμφωνα με τον Πίνακα 4.1, είναι τα 256 bits, ωστόσο χαμηλότερη παρεχόμενη ασφάλεια θα μπορούσε να υποστηριχτεί (π.χ. 128 bits).

- (3) Για να μειώσετε την εντροπία που απαιτείται στον μηχανισμό CTR-DRBG, χρειάζεται απλά στην συνάρτηση `buildCTR_DRBG` να κάνετε την εξής αλλαγή “`.setEntropyBitsRequired(128);`”. Ο λόγος για τον οποίο χτυπάει σφάλμα κατά την εκτέλεση οφείλεται στο γεγονός ότι η ελάχιστη εντροπία που απαιτείται για τον AES-256, σύμφωνα με τον Πίνακα 4.2, είναι τα 256 bits, ωστόσο υψηλότερη εντροπία είναι θεμιτή (π.χ. 1024 bits).

- (4) Η συνάρτηση `buildCTR_3DES_DRBG` θα είχε την εξής μορφή:

```
public static SecureRandom buildCTR_3DES_DRBG(boolean usePerString)
{
    EntropySourceProvider entSource =
        new BasicEntropySourceProvider(new SecureRandom(), true);
    FipsDRBG.Builder drgbBldr = FipsDRBG.CTR_Triple_DES_168
        .fromEntropySource(entSource)
        .setSecurityStrength(112)
        .setEntropyBitsRequired(112);

    if(usePerString){
        drgbBldr.setPersonalizationString(PersonalizationString);
    }
    return drgbBldr.build(Nonce, false);
}
```

και η κλήση της στην συνάρτηση `main` θα γινόταν ως εξής:

```
SecureRandom drgb4 = buildCTR_3DES_DRBG(true);
drgb4.nextBytes(data);
```

4.8.2 (σελ. 93): Οι απαντήσεις στα ερωτήματα αυτής της εργασίας είναι τα εξής:

- (1) Χρειάζεται απλά να αλλάξετε το `bitsize` και συγκεκριμένα να κάνετε αυτή την αλλαγή “`int bitsize = 1024;`”. Ο τρόπος λειτουργίας της γεννήτριας είναι ο ίδιος, αυτό που αλλάζει είναι η πολυπλοκότητα των πράξεων, έχοντας ως αποτέλεσμα βέβαια και μεγαλύτερη παρεχόμενη ασφάλεια.
- (2) Για να παράγετε `Integers` αντί για `BigIntegers` η αλλαγή στον κώδικα που πρέπει να γίνει είναι η εξής:

```
Integer rnd = yar.nextInt();
System.out.println("Decimal Number: "+rnd
    +" \t-> Hex: "+Integer.toHexString(rnd).toUpperCase());
```

B.5 Κεφάλαιο 5

B.5.1 Λύσεις Εργασιών

5.5.1 (σελ. 118): Ενδεικτική λύση της εργασίας είναι διαθέσιμη [εδώ](#).



5.5.2 (σελ. 119): Ενδεικτική λύση της εργασίας είναι διαθέσιμη [εδώ](#).



5.5.3 (σελ. 119): Ενδεικτική λύση της εργασίας είναι διαθέσιμη **εδώ**.



5.5.4 (σελ. 119): Ενδεικτική λύση της εργασίας είναι διαθέσιμη **εδώ**.



B.6 Κεφάλαιο 8

B.6.1 Λύσεις Ασκήσεων

8.6.1 (σελ. 190): Τα κύρια μειονεκτήματα που παρουσιάζει αυτό το πρωτόκολλο, μεταξύ πιθανών άλλων, είναι τα εξής:

- Ο Μπάμπης μπορεί να μάθει ποια είναι η περιουσία της Αλίκης εάν κρατήσει αρχείο του τυχαίους αριθμούς που τοποθέτησε σε όλα τα υπόλοιπα κουτιά.
- Μόνο ο Μπάμπης μπορεί να μαθαίνει ποιος είναι πλουσιότερος. Η Αλίκη μπορεί μόνο να μάθει ποιος είναι πλουσιότερος με βάση το τι θα της πει ο Μπάμπης.

8.6.2 (σελ. 191): Οι απαντήσεις στην άσκηση είναι οι εξής:

- (1) Στο κρυπτοσύστημα RSA το γινόμενο των κρυπτοκειμένων έχει ως αποτέλεσμα το γινόμενο των δύο αριθμών στο αποκρυπτογραφημένο αποτέλεσμα, οπότε:

$$D(E(x_1) \cdot E(x_2)) = D(E(x_1 \cdot x_2)) = x_1 \cdot x_2 = 20 \cdot 4 = 80$$

- (2) Στο κρυπτοσύστημα Paillier το γινόμενο των κρυπτοκειμένων έχει ως αποτέλεσμα το άθροισμα των δύο αριθμών στο αποκρυπτογραφημένο αποτέλεσμα, οπότε:

$$D(E(x_1) \cdot E(x_2)) = D(E(x_1 + x_2)) = x_1 + x_2 = 20 + 4 = 24$$

- (3) Στο κρυπτοσύστημα ElGamal το γινόμενο των κρυπτοκειμένων έχει ως αποτέλεσμα το γινόμενο των δύο αριθμών στο αποκρυπτογραφημένο αποτέλεσμα, οπότε:

$$D(E(x_2) \cdot E(x_2)) = D(E(x_2 \cdot x_2)) = x_2 \cdot x_2 = 4 \cdot 4 = 16$$

8.6.3 (σελ. 191): Τα βήματα που πρέπει να ακολουθηθούν ανάμεσα στον Μπάμπη και στην Αλίκη είναι οι εξής:

1. Η Αλίκη περιμένει στο σημείο 1.
2. Ο Μπάμπης εισέρχεται στην σπηλιά και περιμένει στα σημεία 3 ή 4 (χωρίς η Αλίκη να μπορεί να δει προς τα πάγκες).
3. Όταν ο Μπάμπης δεν είναι πλέον ορατός στην Αλίκη, η Αλίκη προχωρά και περιμένει στο σημείο 2.
4. Η Αλίκη λέει στον Μπάμπη να βγει είτε από τα αριστερά είτε από τα δεξιά στην σπηλιά ανάλογα με την προτίμηση της.
5. Ο Μπάμπης ακολουθεί το πρόσταγμα της Αλίκης και βγαίνει από την πλευρά που του ζητήθηκε χρησιμοποιώντας τον μυστικό κωδικό όποτε είναι αναγκαίο.
6. Η Αλίκη και ο Μπάμπης επαναλαμβάνουν τα βήματα 1 έως 5 λ φορές.

Το παράδειγμα αυτό αποτελεί μια απόδειξη γνώσης μέσω μιας πιθανοτικής διαδικασίας, και συγκεκριμένα, μετά από λ επαναλήψεις, ο Μπάμπης μπορεί να πείσει την Αλίκη πως ξέρει τον μυστικό κωδικό με πιθανότητα $1 - \frac{1}{2^L}$.

B.6.2 Λύσεις Εργασιών

8.6.1 (σελ. 191): Για τη δημιουργία των ασύμμετρων κλειδιών του κρυπτοσυστήματος RSA, να χρησιμοποιήσετε το component του CrypTool 2 με όνομα RSA Key Generator, ενώ για το κρυπτοσύστημα Paillier, να χρησιμοποιήσετε το component με όνομα Paillier Key Generator. Για την ολοκλήρωση της εργασίας θα πρέπει να χρησιμοποιήσετε τις κατάλληλες εξόδους του RSA και Paillier Key Generator τόσο για κρυπτογράφηση όσο και για αποκρυπτογράφηση, και μετά την κρυπτογράφηση των δύο αριθμών να πραγματοποιήσετε το γινόμενο των κρυπτοκειμένων. Ένα ενδεικτικό CrypTool 2 Project που υλοποιεί την ομοιορφική ιδιότητα του RSA είναι διαθέσιμο [εδώ](#), ενώ του Paillier είναι διαθέσιμο [εδώ](#).



8.6.2 (σελ. 192): Οι απαντήσεις στα ερωτήματα αυτής της εργασίας είναι τα εξής:

- (1) Τα αποτελέσματα αποκρυπτογράφησης είναι: 50 – RSA, 50 – ElGamal, και 15 – Paillier.
- (2) Τα αποτελέσματα αποκρυπτογράφησης για τον RSA και τον ElGamal είναι δύο πολύ μεγάλοι αριθμοί γιατί το πεδίο τιμών είναι θετικοί ακέραιοι αριθμοί, ενώ του Paillier είναι 5 γιατί σε αυτή την περίπτωση το αποτέλεσμα συνεχίζει να είναι θετικός αριθμός.
- (3) Ύστερα από την τροποποίηση των πράξεων αποκρυπτογράφησης για τον RSA και τον ElGamal, τα αποτελέσματα διαμορφώνονται ως εξής: -50 – RSA, -50 – ElGamal, και 5 – Paillier.

8.6.3 (σελ. 192): Οι απαντήσεις στα ερωτήματα αυτής της εργασίας είναι τα εξής:

- (1) Το σημείο του κώδικα που προσομοιώνει τα βήματα του πρωτοκόλλου είναι αυτό: “ver.verifyAndSetVjEj(prov.getVjEj_AndSetE(ver.getE_AndSetCuJ(prov.c, prov.getUj())))”.

- (2) Η απόδειξη δεν είναι πλέον έγκυρη γιατί το μήνυμα $m = 2$ δεν ανήκει στο νέο σύνολο τιμών $S = \{0, 1\}$.
- (3) Η απόδειξη είναι έγκυρη γιατί το νέο μήνυμα $m = 1$ ανήκει στο σύνολο τιμών $S = \{0, 1\}$.

8.6.4 (σελ. 192): Για τη δημιουργία των ασύμμετρων κλειδιών του κρυπτοσυστήματος RSA, να χρησιμοποιήσετε το component του CrypTool 2 με όνομα *RSA Key Generator*, για την σύνοψη το component *SHA*, και για τις τυφλές υπογραφές τα components *Blind Signature Generator* και *Blind Signature Verifier*. Για την ολοκλήρωση της εργασίας θα πρέπει να ελέγχετε (έστω οπτικά) εάν η σύνοψη του μηνύματος είναι ίδια με την έξοδο του *Blind Signature Verifier*. Ένα ενδεικτικό CrypTool 2 Project που υλοποιεί τις τυφλές υπογραφές είναι διαθέσιμο [εδώ](#).



ΕΥΡΕΤΗΡΙΟ

- Advanced Encryption Standard (AES), 8
ANSI Retail MAC (AMAC), 101

Bitcoin, 260

Camellia, 11
Carter-Wegman + Counter (CWC), 108
Cipher-based MAC (CMAC), 101
Common Criteria (CC), 145
Counter with CBC-MAC (CCM), 108

EAX Mode, 108
Encrypted CBC-MAC (EMAC), 100
Elliptic Curve Integrated Encryption Scheme – ECIES, 42
Ethereum, 262

Galois/Counter Mode (GCM), 108
Genesis Block, 252

Internet Protocol Security (IPSec), 235
Internet Security Association and Key Management Protocol (ISAKMP), 236
IPSec – Ενθυλάκωση Ωφέλιμου Φορτίου Ασφαλείας, 237
IPSec – Κεφαλίδα Αυθεντικοποίησης, 236
IPSec – Λειτουργία Μεταφοράς, 239
IPSec – Λειτουργία Σήραγγας, 237
IPSec – Συσχετισμός Ασφαλείας, 240

Kerberos, 135

MAC Βασισμένα σε Κρυπτογραφικούς Αλγορίθμους Μπλοκ, 99
MAC Βασισμένα σε Συναρτήσεις Σύνοψης, 103

Rijndael, 9
RSA Probabilistic Signature Scheme (RSA-PSS), 115
RSA-PKCS #1, 114

Secure Shell Protocol (SSH), 243
Serpent, 11

TLS – Πρωτόκολλο Εγγραφής, 229, 234
TLS – Πρωτόκολλο Χειραψίας, 229
Transport Layer Security (TLS), 228

Άγκυρα Εμπιστοσύνης, 151
Έξυπνα Συμβόλαια, 257
Αλγόριθμοι Ελαφράς Κρυπτογραφίας, 313
Αλγόριθμος ASCON, 314
Αλγόριθμος Blum-Blum-Shub, 89
Αλγόριθμος Blum-Micali, 91
Αλγόριθμος GIFT-COFB, 318
Αλγόριθμος SPARKLE (SCHWAEMM και ESCH), 322
Αλγόριθμος TinyJAMBU, 325
Αλγόριθμος Xoodyak, 328
Αλγόριθμος Yarrow, 91
Αλγόριθμος Κρυπτογράφησης, 3
Αλγόριθμος Κρυπτογράφησης Ροής, 18
Αλγόριθμος Συναίνεσης, 257
Αλγόριθμος Ψηφιακών Υπογραφών DSA, 38

- Αλγόριθμος Ψηφιακών Υπογραφών ECDSA, 43
 Αλυσίδα Μπλοκ (Blockchain), 250
 Αλυσίδες Μπλοκ με Άδεια, 253
 Αλυσίδες Μπλοκ χωρίς Άδεια, 254
 Αμεταβλητότητα, 255
 Ανάκτηση Κλειδιού, 127
 Αναδρομική Αποκρυπτογράφηση, 356
 Αναστολή Κλειδιού, 127
 Αποδείξεις Μηδενικής Γνώσης, 182
 Αποκρυπτογράφηση (Decryption), 4
 Απόδειξη Αρχής (PoA), 260
 Απόδειξη Εργασίας (PoW), 257
 Απόδειξη Συμμετοχής (PoS), 258
 Αρχή Εγγραφής (AE), 150, 152
 Αρχή Πιστοποίησης (AP), 150
 Αρχή Πιστοποίησης Ρίζας, 150
 Αρχή του Kerckhoffs, 4
 Αρχικό Κείμενο (Plaintext), 4
 Αρχιτεκτονική των PRNG, 79
 Ασφαλείς Υπολογισμοί Πολλαπλών Οντοτήτων, 173
 Αυθεντικοποιημένη Κρυπτογράφηση, 105
 Αυτο-υπογεγραμμένο Πιστοποιητικό, 151
 Βασιζόμενο Μέρος, 150
 Βυζαντινή Ανοχή Σφαλμάτων (BFT), 259
 Γεννήτριες Τυχαίων Αριθμών, 78
 Δένδρα Merkle, 57
 Δήλωση Πρακτικής Πιστοποίησης (ΔΠΠ), 150
 Δίκτυα Ανωνυμίας, 273
 Δεδομένα Μακροπρόθεσμης Επικύρωσης
 Υπογραφής, 163
 Δεδομένα σε Κίνηση, 225
 Δεδομένα σε Κατάσταση Ηρεμίας, 201
 Δεδομένα σε Μεταφορά, 225
 Δημοπρασίες με Διασφάλιση Ιδιωτικότητας, 281
 Διακλάδωση Αλυσίδας Μπλοκ, 267
 Διαχείριση Κλειδιών, 126
 Διεύρυνση Υπογραφών, 162
 Διπλή Δαπάνη, 268
 Εγκεκριμένη Διάταξη Δημιουργίας
 Ηλεκτρονικής Υπογραφής, 161
 Εγκεκριμένη Ηλεκτρονική Υπογραφή, 160
 Εγκεκριμένο Πιστοποιητικό Ηλεκτρονικής
 Υπογραφής, 160
 Εγκεκριμένος Πάροχος Υπηρεσιών
 Εμπιστοσύνης (ΕΠΥΓΕ), 158
 Εγωιστική Εξόρυξη, 266
 Εκδούσα Αρχή Πιστοποίησης, 151
 Ελαφρά Κρυπτογραφία, 311
 Ελλειπτικές Καμπύλες, 40
 Εξόρυξη Δεδομένων με Διασφάλιση
 Ιδιωτικότητας, 287
 Επίθεση Sybil, 265
 Επίθεση Επέκτασης Μήκους, 104
 Επίθεση του 51%, 266
 Επίπεδο Ασφαλείας, 23
 Επαναλαμβανόμενη Κοινή Χρήση Μυστικών,
 143
 Ετικέτα Αυθεντικοποίησης, 109
 Ηλεκτρονική Σφραγίδα, 165
 Ηλεκτρονική Υπηρεσία Συστημάτων Παράδοσης,
 166
 Ηλεκτρονική Υπογραφή, 159
 Ηλεκτρονική Ψηφοφορία, 276
 Ηλεκτρονικό Πρωτόκολλο Κατάστασης
 Πιστοποιητικού (ΗΠΚΠ), 156
 Κέντρο Διαμοιρασμού Κλειδιών, 136
 Καθυστέρηση Συναίνεσης, 267
 Κανονισμός eIDAS, 157
 Κατασκευή HAIFA, 54
 Κατασκευή Merkle-Damgård, 53
 Κατασκευή Sponge, 55
 Κατασκευή Wide-Pipe, 54
 Κβαντική Δέσμευση, 349
 Κβαντική Διανομή Κλειδιών, 340
 Κβαντική Κοινή Χρήση Μυστικών, 344
 Κβαντική Κρυπτογραφία, 335
 Κβαντική Κρυπτογραφία βάσει Θέσης, 350
 Κβαντική Ρίψη Νομίσματος, 346
 Κλειδί Περιτύλιξης, 131
 Κλειδιά Συνεδρίας, 133
 Κρυπτογράφηση (Encryption), 4
 Κρυπτογράφηση Βάσει Ταυτότητας, 188
 Κρυπτογράφηση Βάσει Χαρακτηριστικών, 189
 Κρυπτογράφηση Δεδομένων σε Επίπεδο Βάσης
 Δεδομένων, 215
 Κρυπτογράφηση Δεδομένων σε Επίπεδο Δίσκου,
 203
 Κρυπτογράφηση Δεδομένων σε Επίπεδο
 Συστήματος Αρχείων, 212
 Κρυπτογραφία Βασισμένη σε Ισογένειες, 373
 Κρυπτογραφία Βασισμένη σε Κώδικα, 372
 Κρυπτογραφία Βασισμένη σε Πλέγματα, 370
 Κρυπτογραφία Ελλειπτικών Καμπυλών, 39
 Κρυπτογραφίας Δημοσίου Κλειδιού, 29
 Κρυπτογραφικός Αλγόριθμος Αντικατάστασης, 4
 Κρυπτογραφικός Αλγόριθμος Αντιμετάθεσης, 5
 Κρυπτογραφικός Αλγόριθμος Μπλοκ, 5

- Κρυπτοκείμενο (Ciphertext), 4
 Κρυπτοπερίοδος, 129
 Κρυπτοσύστημα ElGamal, 34
 Κρυπτοσύστημα Paillier, 36
 Κρυπτοσύστημα RSA, 30
 Κυκλική συνάρτηση, 8
 Κόμβος Επικύρωσης, 253
 Κώδικας Feistel, 7
 Κώδικας Αυθεντικοποίησης Μηνύματος (MAC), 98
 Κώδικας Γινομένου, 8
 Λίστα Ανακληθέντων Πιστοποιητικών (ΛΑΠ), 155
 Μερική Ομομορφική Κρυπτογράφηση, 179
 Μετα-Κβαντική Κρυπτογραφία, 356
 Μηχανική Μάθηση με Διασφάλιση
 Ιδιωτικότητας, 292
 Μηχανισμοί Κρυπτογραφικά Ασφαλών PRNGs, 83
 Μηχανισμός CTR-DRBG, 87
 Μηχανισμός HASH-DRBG, 84
 Μηχανισμός HMAC-DRBG, 85
 Μονάδες Ασφαλείας Υλισμικού (HSM), 144, 161
 Μονοαλφαβητική Αντικατάσταση, 5
 Μπλοκ (Αλυσίδες Μπλοκ), 250
 Οικογένεια Συναρτήσεων Σύνοψης BLAKE, 66
 Οικογένεια Συναρτήσεων Σύνοψης SHA, 61
 Ομαδικές Υπογραφές, 186
 Ομομορφική Κρυπτογράφηση, 177
 Οριθετημένο Μοντέλο Κβαντικής
 Αποθήκευσης, 350
 Ορφανά Μπλοκ, 269
 Πάροχος Υπηρεσιών Εμπιστοσύνης (ΠΥΕ), 150
 Παράμετροι Τομέα ECC, 41
 Παραγωγή Κλειδιού, 127
 Παρωχημένα Μπλοκ, 269
 Πιστοποιητικό Ρίζας, 151
 Πλέγμα (Lattice), 370
 Πλήρης Ομομορφική Κρυπτογράφηση, 180
 Πολιτική Πιστοποίησης (ΠΠ), 150
 Πολυαλφαβητική Αντικατάσταση, 5
 Πολυμεταβλητή Κρυπτογραφία, 374
 Πρακτική Βυζαντινή Ανοχή Σφαλμάτων
 (PBFT), 259
 Προηγμένη Ηλεκτρονική Υπογραφή, 159
 Προωθημένη Μυστικότητα, 133
 Πρωτόκολλα Εδραίωσης Κλειδιών, 132
 Πρωτόκολλα Μεταφοράς Κλειδιών, 127
 Πρωτόκολλα Συμφωνίας Κλειδιών, 127
 Πρωτόκολλο Συμφωνίας Κλειδιού, 133
 Συνάρτηση Σύνοψης, 51
 Συνάρτηση Σύνοψης BLAKE, 67
 Συνάρτηση Σύνοψης BLAKE2, 69
 Συνάρτηση Σύνοψης BLAKE3, 70
 Συνάρτηση Σύνοψης MD5, 59
 Συνάρτηση Σύνοψης SHA-1, 62
 Συνάρτηση Σύνοψης SHA-2, 63
 Συνάρτηση Σύνοψης SHA-3, 65
 Συνάρτηση Σύνοψης Whirlpool, 60
 Συναλλαγή (Αλυσίδες Μπλοκ), 252
 Σχήμα Κοινής Χρήσης Μυστικών
 Ito-Nishizeki-Saito, 142
 Σχήματα Διάσπασης Κλειδιού, 140
 Σχήματα Κοινής Χρήσης Μυστικών, 140
 Σύστημα Lamport, 359
 Σύστημα Winternit, 361
 Σύστημα WOTS+, 361
 Σύστημα Διαχείρισης Κρυπτογραφικών
 Κλειδιών (ΣΔΚΚ), 145
 Τεχνολογία Κατανεμημένου Καθολικού (DLT), 250
 Το Πρωτόκολλο BB84, 341
 Το Πρωτόκολλο E91, 343
 Το Πρωτόκολλο του Yao (Garbled Circuit), 174
 Τρόπος Λειτουργίας (Operation Mode), 11
 Τρόπος Λειτουργίας Αλυσιδωτού Μπλοκ
 (CBC), 13
 Τρόπος Λειτουργίας Ανατροφοδότησης
 Κρυπταλγορίθμου (CFB), 16
 Τρόπος Λειτουργίας Ηλεκτρονικού Βιβλίου
 Κωδικών (ECB), 11
 Τρόπος Λειτουργίας Μετρητή (CTR), 15
 Τυφλές Υπογραφές, 185
 Τυχαίοι Αριθμοί, 78
 Υπηρεσίες Εμπιστοσύνης, 158
 Υπογραφές Βασισμένες σε Συνόψεις (HBS), 358
 Υπογραφές Δακτυλίου, 187
 Υπογραφές μιας Χρήστης, 359
 Υποδομή Δημοσίου Κλειδιού (ΥΔΚ), 149
 Υποκείμενη Αρχή Πιστοποίησης (ΥΑΠ), 151
 Ψηφιακά Πιστοποιητικά, 152
 Ψηφιακή Ταυτότητα, 168
 Ψηφιακή Υπογραφή με Ανάκτηση Μηνύματος, 112
 Ψηφιακή Υπογραφή με Προσθήκη, 112
 Ψηφιακό Πορτοφόλι, 168

Θεμελιώσεις και Εφαρμογές της Σύγχρονης Κρυπτογραφίας

ΚΑΛΛΙΠΟΣ 2024
ISBN 978-618-85370-X-X