Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή
Στην Ασφάλεια Υπολογιστών και Δικτύων

ΚΒΑΝΤΙΚΗ ΚΡΥΠΤΟΓΡΑΦΙΑ : «ΤΡΕΧΟΥΣΑ ΚΑΤΑΣΤΑΣΗ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΚΑΤΕΥΘΥΝΣΕΙΣ»

Βιολέττα Μακρή

Επιβλέπων Καθηγητής
Κωνσταντίνος Λιμνιώτης

Μάϊος 2023

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

ΚΒΑΝΤΙΚΗ ΚΡΥΠΤΟΓΡΑΦΙΑ : «ΤΡΕΧΟΥΣΑ ΚΑΤΑΣΤΑΣΗ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΚΑΤΕΥΘΥΝΣΕΙΣ»

Βιολέττα Μακρή

Επιβλέπων Καθηγητής
Κωνσταντίνος Λιμνιώτης

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στην Ασφάλεια Υπολογιστών και Δικτύων

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Μάϊος 2023

Περίληψη

Η παρούσα μεταπτυχιακή διατριβή έχει ως θέμα έναν από τα πιο εκρηκτικά εξελισσόμενα πεδία του σύγχρονου κόσμου: τη κβαντική κρυπτογραφία. Αποτελεί πεδίο τομής πολλών γνωστικών πεδίων οπότε κρίνεται απαραίτητη η αποσαφήνιση πολλών εννοιών άλλων γνώσεων που είναι απαραίτητα προαπαιτούμενα για τη κατανόησή της. Για να επιτευχθεί αυτή λοιπόν, θα δωθούν απαραίτητοι μαθηματικοί ορισμοί (κυρίως της άλγεβρας και θεωρία αριθμών) που αποτέλεσαν ισχυρή βάση για τη δημιουργία γενικότερα της κρυπτογραφίας. Με αυτά τα πολύτιμα μαθηματικά εργαλεία, θα μελετηθεί γενικότερα η κρυπτογραφία μαζί με τα πρώτα σημαντικά πρωτόκολλα που χρησιμοποιούνται μέχρι και σήμερα για την ασφάλεια πολλαπλών μορφών επικοινωνίας. Εξίσου σημαντικό γνωστικό πεδίο αποτελεί αυτό της κβαντικής φυσικής που αποτέλεσε συνδετικό κρίκο στο πέρασμα από τη κλασική κρυπτογραφία στο κεντρικό θέμα της διατριβής: τη κβαντική κρυπτογραφία. Αφού αποσαφηνιστεί το θεωρητικό υπόβαθρο γενικότερα της κβαντικής κρυπτογραφίας, έπειτα θα αναλυθούν οι άρχες τριών ιδιαίτερα γνωστών πρωτοκόλλων αυτής: του BB84, του B92 και του E91. Θα πραγματοποιηθεί εφαρμογή και των δύο με χρήση κώδικα προς εξαγωγή συμπερασμάτων και έπειτα θα συζητηθούν (σύντομα) άλλα πρωτόκολλα που αναδύθηκαν ανά τα χρόνια στη προσπάθεια βελτίωσης των προηγούμενων. Τέλος, θα γίνει σύνοψη και αναφορά σε μελλοντικές προοπτικές και προκλήσεις.

Abstract

The subject of this master's thesis is one of the most explosively evolving fields of the modern world: quantum cryptography. It is a field of intersection of many cognitive fields, so it is necessary to clarify many concepts of other knowledge that are necessary prerequisites for its understanding. In order to achieve this, necessary mathematical definitions (mainly algebra and number theory) will be given that formed a strong basis for the creation of cryptography in general. With these valuable mathematical tools, cryptography in general will be studied along with the first important protocols used to this day for the security of multiple forms of communication. An equally important field of knowledge is that of quantum physics, which was a connecting link in the transition from classical cryptography to the central topic of the thesis: quantum cryptography. After clarifying the theoretical background of quantum cryptography in general, the origins of three particularly well-known protocols will be analyzed: BB84, B92 and E91. An implementation of them using inferred code will be carried out, and then other protocols that have emerged over the years in an attempt to improve upon the previous ones will be discussed (shortly). Finally, there will be a summary and reference to future perspectives and challenges.

Ευχαριστίες

Θα ήθελα να εκφράσω τις ευχαριστίες μου στον επιβλέποντα καθηγητή μου κύριο Λιμνιώτη για τη καθοδήγηση και τις πολύτιμες συμβουλές που μου προσέφερε κατά τη διάρκεια του δύσκολου έργου της εκπόνησης αυτή της μεταπτυχιακής διατριβής.

Επίσης, ευχαριστώ θερμά την οικογένειά μου και τους κοντινούς μου φίλους για την ηθική συμπαράσταση, στήριξη και κατανόηση που μου έδειξαν σε όλη τη διάρκεια των σπουδών μου.

Περιεχόμενα

Κατάλογος εικόνων

Κεφάλαιο 1
Εισαγωγή

Για χιλιάδες χρόνια τα ανθρώπινα όντα χρησιμοποιούν κώδικες ώστε να διατηρούν μυστικά. Στη σύγχρονη εποχή όλο και περισσότεροι άνθρωποι χρησιμοποιούν δικτυωμένα συστήματα ώστε να επικοινωνούν μεταξύ τους. Με την άνοδο του διαδικτύου τα ευαίσθητα προσωπικά μας οικονομικά δεδομένα και δεδομένα υγείας, καθώς και εμπορικά και κρατικά μυστικά σε ευρύτερο επίπεδο, μεταδίδονται τακτικά μέσω του διαδικτύου[42].

Ωστόσο, υπάρχουν πολλά μειονεκτήματα στη χρήση αυτού του τύπου επικοινωνίας λόγω του ότι επιτιθέμενοι προσπαθούν να αποκτήσουν πληροφορίες για τα προσωπικά δεδομένα των χρηστών κατά τη διάρκεια της επικοινωνίας, καθιστώντας την ανασφαλή. Αυτή ήταν η αιτία για τη γέννηση της αναγκαιότητας ασφάλειας εξουσιοδότησης των ατόμων που απαρτίζουν το εκάστοτε δίκτυο. Ως αποτέλεσμα, αναπτύχθηκε η κρυπτογραφία, με τα πρωτόκολλα διανομής κλειδιών που εμπεριέχονται σε αυτή. Όπως υποδηλώνει το όνομά τους στόχος είναι η ασφαλής κοινοποίηση ενός ή περισσότερων κλειδιών μέσω του καναλιού επικοινωνίας, πριν ξεκινήσει η ανταλλαγή πληροφοριών μεταξύ των χρηστών[41].

Ωστόσο, όλες οι κρυπτογραφικές τεχνικές θα είναι αναποτελεσματικές αν ο μηχανισμός διανομής του κλειδιού θα είναι αδύναμος . Η ασφάλεια των πιο σύγχρονων κρυπτογραφικών συστημάτων κλειδιού βασίζεται στην υπολογιστική πολυπλοκότητα και στον εξαιρετικά μεγάλο χρόνο που απαιτείται ώστε να σπάσει ο κώδικας[33].

1.1 Αντικείμενο

Με τις εξαιρετικά σημαντικές ανακαλύψεις του 20ου αιώνα στο πεδίο της κβαντομηχανικής αναπτύχθηκε μία εξελιγμένη μορφή κρυπτογραφίας η κβαντική κρυπτογραφία, που έμελλε να αποτελέσει ένα από τα σημαντικότερα πεδία στο σύγχρονο κόσμο της ασφάλειας πληροφοριών. Αποτελεί εξαιρετικά πρόσφατη τεχνολογία η οποία εξελίχθηκε λόγω ελαττωμάτων στα κλασικά κρυπτογραφικά συστήματα[36]. Θεμελιώδεις νόμοι της κβαντικής φυσικής εφαρμόζονται από τη κβαντική κρυπτογραφία για την εγγύηση ασφαλούς επικοινωνίας μεταξύ νόμιμων χρηστών. Η κβαντική κρυπτογραφία αποτελεί ένα ευρύ γνωστικό πεδίο που εμπεριέχει μεγάλο φάσμα κρυπτογραφικών πρακτικών και πρωτοκόλλων, και όπως είναι αναμενόμενο υπάρχει χώρος για σημαντική έρευνα σχετικά με αυτό προς δημιουργία νέων πρακτικών. Αυτή η διατριβή λοιπόν, αποτελεί μία βιβλιογραφική επισκόπηση σχετικά με τη κβαντική κρυπτογραφία και των σύγχρονων μεθόδων της.

1.2 Ερευνητικά Ερωτήματα

Τα κύρια ερευνητικά ερωτήματα με τα οποία καταπιάνεται αυτή η διατριβή αποτελούν η ασφάλεια μετάδοσης των μηνυμάτων μέσω κρυπτογραφικών πρωτοκόλλων, η αξιολόγησή τους με τους απαραίτητους δείκτες, η σύγκριση των κρυπτογραφικών πρωτοκόλλων μεταξύ τους, καθώς και η παράδοση κατευθυντήριων γραμμών για νέα ερευνητικά ευρήματα προς τους επόμενους επιστήμονες που θα τη μελετήσουν.

1.3 Μεθοδολογία

Σχετικά με τη μεθοδολογία που ακολουθήθηκε, αυτή ήταν η βιβλιογραφική επισκόπηση μέσα από βιβλιογραφικές πηγές τόσο διαχρονικές όσο και σύγχρονες για την εξακρίβωση των SOTA τεχνικών. Η πειραματική ρύθμιση περιλαμβάνει υλοποίηση των πρωτοκόλλων (που αναλύθηκαν πρώτα θεωρητικά) σε κώδικα του αποθετηρίου github [30,49] σε περιβάλλον Google Colab. Τα πρωτόκολλα που υλοποιήθηκαν αποφασίστηκε να είναι τα συγκεκριμένα με βάση την ευρεία χρήση τους σε ζητήματα κβαντικής κρυπτογραφίας. Η παραγωγή των bits για την διεξαγωγή των πειραμάτων έγινε με χρήση γεννητριών ψευδοτυχαίων αριθμών.

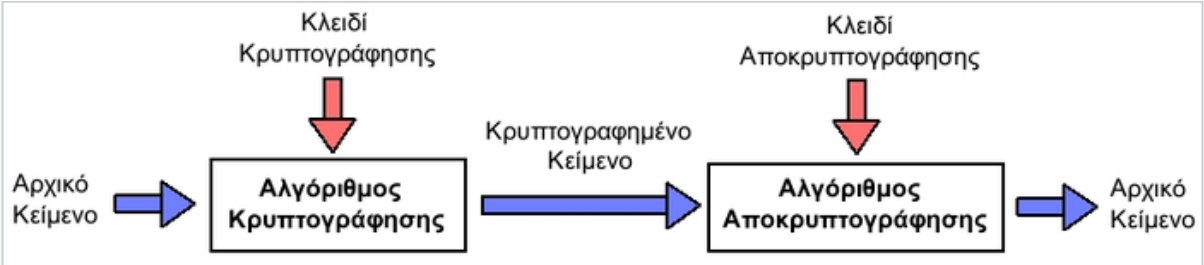
1.4 Δομή

Όσο αφορά τη δομή, αρχικά γίνεται μια σύντομη εισαγωγή στο θέμα και τη σημαντικότητα που διαδραματίζει στο σύγχρονο επιστημονικό γίγνεσθαι. Στο δεύτερο κεφάλαιο γίνεται εισαγωγή στη κρυπτογραφία και το έντονο μαθηματικό υπόβαθρο που τη συνοδεύει. Στο τρίτο αναλύεται η κβαντική φυσική που είναι απαραίτητη για τη μετάβαση από τη κρυπτογραφία στη κβαντική κρυπτογραφία που εισάγεται στο τέταρτο κεφάλαιο, όπου δίνονται τα πρωτόκολλά της και τα πειράματα που τα συνοδεύουν. Στο πέμπτο και τελευταίο κεφάλαιο δίνονται οι κατευθύνσεις/προκλήσεις προς μελλοντική έρευνα και ο επίλογος.

Κεφάλαιο 2
Κρυπτογραφία

2.1 Εισαγωγή στην Κρυπτογραφία

Η λέξη κρυπτογραφία, προέρχεται από τα συνθετικά «κρύπτος» και «γράφω» και αποτελεί ένα διεπιστημονικό γνωστικό πεδίο (τομή των επιστημών των μαθηματικών, της φυσικής, της επιστήμης υπολογιστών κ.α.) και αποτελεί έναν από τους δύο κλάδους της κρυπτολογίας (ο άλλος είναι η κρυπτανάλυση), η οποία ασχολείται με τη μελέτη της ασφαλούς επικοινωνίας. Η κρυπτογραφία ασχολείται με την ανάπτυξη και χρήση τεχνικών κρυπτογράφησης και αποκρυπτογράφησης, με σκοπό την απόκρυψη του περιεχομένου των μηνυμάτων από κάποιον τρίτο ο οποίος δεν διαθέτει εξουσιοδότηση. Με τον όρο κρυπτογράφηση -ηδη πριν τη σύγχρονη εποχή- εννοούμε τη μετατροπή αναγνώσιμων πληροφοριών (απλό κείμενο) σε ένα ανούσια κείμενο ακατάληπτης φύσης (κρυπτογραφημένο κείμενο γνωστό με την ονομασία ciphertext). Το τελευταίο μπορεί να διαβαστεί μόνο με αντιστροφή της διαδικασίας, δηλαδή αποκρυπτογράφηση. Για την προηγούμενη διαδικασία σημαντική είναι η έννοια του κλειδιού που είναι ένας αριθμός αρκετών ψηφίων (bits) που χρησιμοποιείται ως είσοδος στη συνάρτηση κρυπτογράφησης:



Εικόνα 2.1: Τυπικό σύστημα κρυπτογράφησης-απορυπτογράφησης.

Γενικά ισχύει ο κανόνας ότι το μέγεθος του κλειδιού κρυπτογράφησης, είναι ανάλογο της εμπιστευτικότητας του κρυπτογραφημένου μηνύματος. Πρακτικές εφαρμογές βρισκει στο ηλεκτρονικό εμπόριο, τα ψηφιακά νομίσματα, του κωδικούς πρόσβασης υπολογιστών και τις στρατιωτικές επικοινωνίες. Ιδιαίτερα στις τελευταίες έμελλε να διαδραματίσει ουσιαστικό ρόλο καθώς συνεισέφερε στη διασφάλιση του απορρήτου μεταξύ στρατιωτικών ηγετών, κατασκόπων και διπλωματών.

Στόχοι της κρυπτογραφίας αποτελούν:

- Εμπιστευτικότητα δεδομένων: Η κατανόηση της πληροφορίας μόνο μεταξύ εξουσιοδοτημένων μελών.

- Ακεραιότητα δεδομένων: Η αλλοίωση της πληροφορίας να συμβαίνει μόνο από τα εξουσιοδοτημένα μέλη και να αδυνατεί να αλλοιωθεί απο τρίτους χωρίς να γίνει αντιληπτή.

- Πιστοποίηση: Η διαβεβαίωση ότι οι ταυτότητες του αποστολέα και του παραλήπτη δεν είναι πλαστές, καθώς και η πηγή και προορισμός της πληροφορίας.

- Μη Άρνηση: Η μη άρνηση της αυθεντικότητας μετάδοσης ή της δημιουργίας της πληροφορίας από τον αποστολές ή τον παραλήπτη.

Γενικά, Οι κρυπτογραφικοί αλγόριθμοι σχεδιάζονται γύρω από υποθέσεις υπολογιστικής στιβαροτητας, καθιστώντας τέτοιους αλγόριθμους δύσκολο να παραβιαστούν στην πραγματική πρακτική από οποιονδήποτε αντίπαλο. Δηλαδή, υπάρχει θεωρητικά η δυνατότητα κάποιος να επιτεθεί στο σύστημα, ακόμα και αν αυτό είναι καλά σχεδιασμένο, αλλά είναι ανέφικτο στη πράξη. Τέτοια συστήματα, εάν έχουν σχεδιαστεί καλά, ονομάζονται

επομένως "υπολογιστικά ασφαλή". Ωστόσο, η εκρηκτική εξέλιξη της τεχνολογίας υπολογιστών, δημιουργεί την ανάγκη επαναθεμελίωσης νέων συστημάτων για περαιτέρω υπολογιστική ασφάλεια. Για αυτό η κρυπτογραφία ιστορικά πέρασε πολλές περιόδους κατά τις οποίες αποδομούνταν προηγούμενα συστήματα και δημιουργούνταν άλλα νέα.

2.2 Ιστορική αναδρομή

2.2.1 Πρώτη περίοδος κρυπτογραφίας

Χρονολογείται από το 1900 π.Χ. έως το 1900 μ.Χ. και κατά τη διάρκεια αυτής της περιόδου αναπτύχθηκε πληθώρα μεθόδων κρυπτογράφησης, οι οποίες βασίζονταν κυρίως στην επινοητικότητα των δημιουργών τους, χωρίς χρήση πολύπλοκων συσκευών ή εξειδικευμένων γνώσεων.Μερικά παραδείγματα αυτής της εποχής αποτελούν:

- Η σφηνοειδής επιγραφή : Αναπτύχθηκε από λαούς της Μεσοποταμίας που φαίνεται να ασχολήθηκαν με τη κρυπτογραφία ήδη από το 1500 π.Χ.,γεγονός που τη χαρακτηρίζει ως το αρχαιότερο σύστημα κρυπτογραφίας.

- Η «σκυτάλη» : Αναπτύχθηκε από τους Σπαρτιάτες κατά τον πέμπτο αιώνα π.Χ. και αποτέλεσε τη πρώτη στρατιωτική χρήση της κρυπτογραφίας. Ήταν μια ξύλινη ράβδος, ορισμένης διαμέτρου, γύρω από την οποία ήταν τυλιγμένη ελικοειδώς μια λωρίδα περγαμηνής. Το κείμενο ήταν γραμμένο σε στήλες, ένα γράμμα σε κάθε έλικα, όταν δε ξετύλιγαν τη λωρίδα, το κείμενο ήταν ακατάληπτο εξαιτίας της αναδιάταξης των γραμμάτων. Το «κλειδί» ήταν η διάμετρος της σκυτάλης.

- Κρυπτοσύστημα Καίσαρα: Αποτέλεσε το πρώτο σύστημα γραπτής αντικατάστασης γραμμάτων. Πιο συγκεκριμένα, κάθε γράμμα αντιστοιχείται σε εκείνο που βρίσκεται 3 θέσεις μακριά του π.χ. $\boldsymbol{a} \rightarrow \boldsymbol{d}, \boldsymbol{b} \rightarrow \boldsymbol{e}$ κλπ.

2.2.2 Δεύτερη Περίοδος Κρυπτογραφίας

Χρονολογείται μεταξύ 1900 και 1950. Οι δύο παγκόσμιοι πόλεμοι αυτης της περιόδου δημιούργησαν τεράστια ανάγκη για τη μετάδοση ζωτικής σημασίας πληροφοριών μεταξύ στρατευμάτων. Αυτή ώθησε τους επιστήμονες της εποχής να δομήσουν νέα πολυπλοκότερα κρυπτογραφικά συστήματα. Χαρακτηριστικό που τη διαχωρίζει από τη προηγούμενη περίοδο αποτελεί η εισαγωγή ηλεκτρομηχανικών κατασκευών. Αντιπροσωπευτικότερο παράδειγμα αυτών αποτελεί η μηχανή Enigma που αναπτύχθηκε στη Γερμανία. Παρά την αρχική της επιτυχία, μπόρεσε να αποκρυπτογραφηθεί από έναν υπολογιστή που κατόρθωσαν να δημιουργήσουν Βρετανοί επιστήμονες (που έφερε το όνομα Colossus).

2.2.3 Τρίτη Περίοδος Κρυπτογραφίας

Αυτή η περίοδος (1950 μέχρι σήμερα) χαρακτηρίζεται από την έξαρση της ανάπτυξης στους επιστημονικούς κλάδους των μαθηματικών, της μικροηλεκτρονικής και των υπολογιστικών συστημάτων. Αυτή τη περίοδο υπήρξε πληθώρα εργασιών επάνω στη θεωρία δεδομένων και επικοινωνίας,οι οποίες καθιέρωσαν μια στερεά θεωρητική βάση για την κρυπτογραφία και την κρυπτανάλυση. Εκείνη την εποχή η κρυπτογραφία εξαφανίζεται και φυλάσσεται από τις μυστικές υπηρεσίες κυβερνητικών επικοινωνιών όπως η NSA. Η τελευταία ήταν και η πρώτη εθνική υπηρεσία η οποία ενέκρινε έναν δημόσιο προσιτό αλγόριθμο ο οποίος ήταν ο DES (Data Encryption Standard). Αυτός αντικαταστήθηκε επίσημα από τον AES (Advanced Encryption Standard) το 2001 ο οπίος αποτέλεσε σημαντική βελτίωση του προηγούμενου.[06-08]

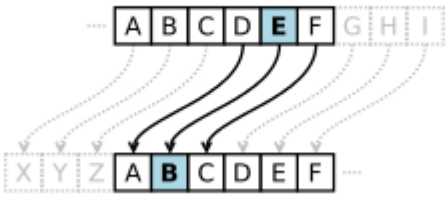
2.3 Κλασικά Κρυπτοσυστήματα

Οι δύο βασικές τεχνικές κρυπτογράφησης που διακρίνουν τα κλασικά κρυπτοσυστήματα αποτελούν την αντικατάσταση και την αντιμετάθεση χαρακτήρων

2.3.1 Σύστημα αντικατάστασης

Ένας αλγόριθμος αντικατάστασης είναι μια μέθοδος κρυπτογράφησης κατά την οποία μονάδες απλού κειμένου (μπορεί να είναι χαρακτήρες με διάφορους συνδυασμούς) αντικαθίσταται από μονάδες κρυπτογραφημένου κειμένου, βάση κάποιων κανόνων

συστήματος. Με την ακριβώς αντίστροφη διαδικασία ο παραλήπτης δύναται να αποκωδικοποιήσει το σύστημα. Ένα τέτοιο παράδειγμα είναι και ο αλγόριθμος του Καίσαρα που αναφέρθηκε προηγουμένως στην ιστορική αναδρομή. Εκεί όπως είδαμε τα γράμματα μετατοπίζονται κατά τρία προς τη μία κατεύθυνση για κρυπτογράφηση και κατά τρία προς την αντίστροφη για αποκρυπτογράφηση:



Εικόνα 2.2 : Οπτικοποίηση της διαδικασίας του αλγορίθμου του Καίσαρα

Δηλαδή μιλάμε για απλώς μία πρόσθεση ενός κλειδιού k με $\boldsymbol{k} \in \mathbb{Z}_{\boldsymbol{24}}$ για ελληνικό αλφάβητο ή $\boldsymbol{k} \in \mathbb{Z}_{\boldsymbol{26}}$ για λατινικό. Σε αυτούς τους αλγορίθμους κάθε γράμμα αντιστοιχεί σε έναν αριθμό δηλαδή $\boldsymbol{A \to 0, B \to 1, \dots, \Omega \to 23}$. Για παράδειγμα με αυτόν τον τρόπο το μήνυμα «ΚΥΠΡΟΣ» μπορεί να κρυπτογραφηθεί για $\boldsymbol{k = 3}$ ως «ΝΨΤΥΣΦ». Ενώ από το τελευταίο με αφαίρεση του $\boldsymbol{k = 3}$ εύκολα μπορούμε να αποκρυπτογραφήσουμε και να καταλήξουμε ξανά στο μήνυμα «ΚΥΠΡΟΣ». Υπάρχει μια πληθώρα τύπων αλγόριθμων αντικατάστασης. Αν ο αλγόριθμος ασχολείται με απλούς χαρακτήρες τότε ονομάζεται απλός αλγόριθμος αντικατάστασης, ενώ αν διαχειρίζεται μεγαλύτερες ομάδες χαρακτήρων ονομάζεται πολυγραφικός. Είναι προφανές ότι ένα κρυπτοσύστημα που χρησιμοποιεί τον αλγόριθμο αυτό είναι αρκετά ευάλωτο αφού ο αντίπαλος μπορεί απλά να δοκιμάσει κάθε δυνατή τιμή του κλειδιού μέχρι να βρεθεί μία λέξη με νόημα και να έπειτα να σπάσει ολόκληρο το μήνυμα και το σύστημα.

2.3.2 Σύστημα αντιμετάθεσης

Σε ένα σύστημα αντιμετάθεσης οι μονάδες απλού κειμένου αναδιατάσσονται κατά έναν διαφορετικό και συνήθως αρκετά περίπλοκο τρόπο, χωρίς να αλλοιώνονται οι μονάδες. Διαφοροποιείται από τους αλγόριθμους αντικατάστασης όπου οι μονάδες απλού κειμένου παραμένουν άθικτες στην ίδια σειρά στο κρυπτογράφημα, αλλάζοντας όμως οι μονάδες αναμεταξύ τους. Αρχικά, το καθαρό κείμενο θα τοποθετηθεί σε έναν πίνακα. Από κάθε γραμμή λαμβάνονται τα γράμματα που αποτελούν το κρυπτογραφημένο κείμενο με διαφορετική

σειρά από αυτή που γράφονται στο καθαρό κείμενο. Αυτό συμβάλλει στην αναδιάταξη των γραμμάτων του καθαρού κειμένου για την παραγωγή του κρυπτογραφήματος. Το κλειδί, σε αυτήν την περίπτωση, είναι η σειρά με την οποία λήφθηκαν τα κρυπτογραφημένα σύμβολα και ο αριθμός των στηλών του πίνακα. Ένας τρόπος με τον οποίο μπορεί να καθοριστεί το κλειδί είναι χρησιμοποιώντας κώδικες λέξεις ή φράσεις των οποίων τα γράμματα καθορίζουν τη σειρά ανάλογα με τη θέση τους στην αλφάβητο. Τα κρυπτοσυστήματα αντιμετάθεσης δεν αποτελούν ασφαλή κρυπτοσυστήματα. Υπάρχουν μόλις *n* δυνατότητες για το κλειδί και σε πολλές περιπτώσεις το *n* δεν είναι μεγάλος αριθμός, με αποτέλεσμα το κλειδί και κατά συνέπεια το καθαρό κείμενο να μπορούν να βρεθούν εύκολα δοκιμάζοντας όλες τις περιπτώσεις. Συνεπώς, αυτή η μέθοδος κρυπτογράφησης καθίσταται εξαιρετικά απλή και θα πρέπει να συνδιάζεται με κάποια άλλη ιδέα.

2.4 Σύγχρονα Κρυπτοσυστήματα

Με την ραγδαία τεχνολογική ανάπτυξη του εικοστού αιώνα η κρυπτογραφία γνώρισε τεράστια πρόοδο ώστε να καλύψει τις αδυναμίες των τεχνικών κλασικής κρυπτογραφίας. Δημιουργήθηκαν νέες πολυπλοκότεροι μέθοδοι κρυπτογραφίας προς τη δόμηση ισχυρότερης άμυνα έναντι κακόβουλων επιθέσεων. Τη κύρια συμβολή σε αυτό το γεγονός την παρείχε η δημιουργία του πρώτου υπολογιστή. Έτς, οι σύγχρονες τεχνικές κρυπτογράφησης σχεδιάστηκαν ειδικά για τη χρήση τους από ηλεκτρονικούς υπολογιστές και χρησιμοποίησαν τα bits, αντί για το αλφάβητο που είχε κάθε γλώσσα όπως είδαμε στις κλασσικές τεχνικές κρυπτογράφησης. Οι σύγχρονες τεχνικές κρυπτογράφησης μπορούν να διαιρεθούν σε δύο κατηγορίες: την συμμετρική κρυπτογράφηση και την ασύμμετρη κρυπτογράφηση.[09-11]

2.4.1 Συμμετρική κρυπτογραφία

Η συμμετρική κρυπτογραφία (ή εναλλακτικά ιδιωτικού κλειδιού) αναφέρεται σε μεθόδους κρυπτογράφησης στις οποίες τόσο ο αποστολέας όσο και ο παραλήπτης μοιράζονται το ίδιο κλειδί (ή, λιγότερο συχνά, στις οποίες τα κλειδιά τους είναι διαφορετικά, αλλά σχετίζονται με έναν εύκολα υπολογίσιμο τρόπο). Αυτό ήταν το μόνο είδος κρυπτογράφησης που ήταν δημοσίως γνωστό μέχρι τον Ιούνιο του 1976[08]. Τα ονόματα που θα χρησιμοποιούνται για όλα τα σενάρια επικοινωνίας που θα ακολουθήσουν θα είναι Alice, Bob (τα άτομα που προσπαθούν να επικοινωνήσουν) και Eve (η «λαθρακροάτρια» που προσπαθεί να εισχωρήσει κρυφά στο σύστημα επικοινωνίας). Αυτά είναι τα ονόματα που

χρησιμοποιούσαν συμβατικά οι επιστήμονες όταν εξέταζαν σενάρια κρυπτογραφίας. Τα στάδια της επικοινωνίας του συμμετρικού κρυπτοσυστήματος είναι τα ακόλουθα:

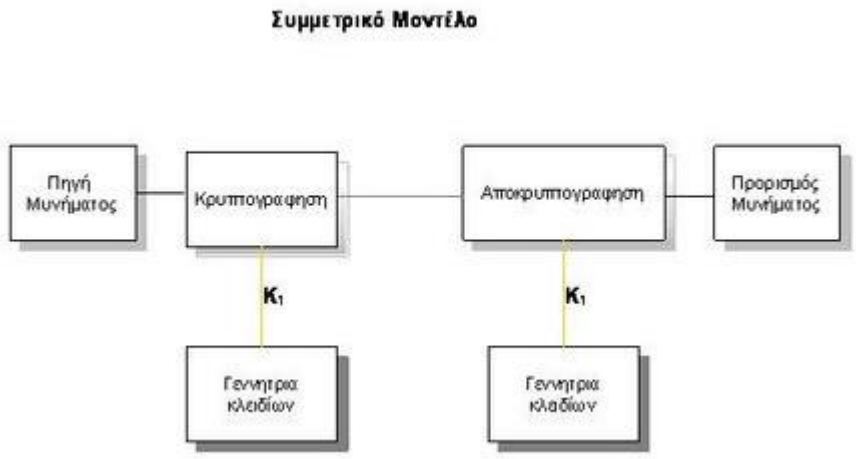
1) Ο Bob ή η Alice αποφασίζει για ένα κλειδί, το οποίο το επιλέγει τυχαία μέσα από ένα ευρύ φάσμα δυνατών τιμών που ονομάζεται κλειδοχώρος (keyspace).

2) Η Alice αποστέλλει το κλειδί στον Bob μέσω ενός ασφαλούς καναλιού.

3) Ο Bob δημιουργεί ένα μήνυμα όπου τα σύμβολα ανήκουν στο χώρο των μηνυμάτων.

4) Το κρυπτογραφεί με το κλειδί που παρέλαβε και αποστέλει την προκύπτουσα κρυπτοσυμβολοσειρά στην Alice.

5) Η Alice λαμβάνει την κρυπτοσυμβολοσειρά και στη συνέχεια με το ίδιο κλειδί την αποκρυπτογραφεί και η έξοδος που παράγεται είναι το μήνυμα [09],[12]:



Εικόνα 2.3 : Μοντελοποίηση συμμετρικού κρυπτοσυστήματος

DES

Ο κρυπτογραφικός αλγόριθμος Data Encryption Standard (DES) είναι ένας συμμετρικός αλγόριθμος δέσμης, που αποτελεί τη πεμπτουσία των αλγορίθμων τμημάτων, καθώς θεωρήθηκε το πρώτο πρότυπο πρακτικά ασφαλούς συστήματος απόκρυψης[13-15]. Ο αλγόριθμος DES έχει σχεδιαστεί για να κρυπτογραφεί το μήνυμα σε blocks δεδομένων που αποτελούνται από 64bits με την χρήση ενός κλειδιού μεγέθους 64bits (56bits το κλειδί και 8

parity bits), όπου το ίδιο ισχύει και για την αποκρυπτογράφηση,η οποία επιτυγχάνεται χρησιμοποιώντας το ίδιο κλειδί όπως στην κρυπτογράφηση. Με την έννοια του «parity bits» εννοούμε ότι αυτά χρησιμοποιούνται αποκλειστικά για τον έλεγχο ισοτιμίας. Έτσι, στη πραγματικότητα τα υπόλοιπα 56 είναι τα bits χρησιμοποίησης του αλγορίθμου. Επιπλέον, αποτελείται από 16 γύρους που ονομάζονται "Δίκτυο Feistel". είναι αρχετυπικός block cipher, δηλαδή, ένας πρωτότυπος κρυπταλγόριθμος συμμετρικού κλειδιού, που λαμβάνει μια σειρά από bits απλού κειμένου (plaintext bits) σταθερού μήκους και την μετατρέπει, μέσω μιας σειράς πολύπλοκων ενεργειών, σε μια άλλη σειρά bits, το κρυπτοκείμενο (chiphertext) με το ίδιο μήκος. Οι βασικοί τρόποι λειτουργίας του DES και των παραλλαγών του, είναι τέσσερις, και έχουν ως στόχο την μεγαλύτερη ασφάλεια του και την απόκρυψη «υπολειμμάτων» του plaintext στο ciphertext. Κάθε φορά επιλέγουμε τον τρόπο με τον οποίο θα λειτουργήσει ο αλγόριθμος. Οι τρόποι λειτουργίας αυτού είναι[03],[14-16]:

- ECB (Electronic Code Book): Στη μέθοδο ηλεκτρονικού βιβλίου κωδικών (ECB) κάθε ομάδα απλού κειμένου κρυπτογραφείται ανεξάρτητα από το κρυπτογράφημα ομάδας και η κρυπτογράφηση εξαρτάται από το κλειδί. Όπως φαίνεται και από το όνομα του, δημιουργείται ένα ηλεκτρονικό βιβλίο, με μοναδική εγγραφή για κάθε block αρχικού κειμένου. Τα δεδομένα είναι χωρισμένα σε 64-bit μπλοκ και κάθε μπλοκ είναι κρυπτογραφημένο, ένα κάθε φορά. Επιτελείται ξεχωριστή κρυπτογράφηση με διαφορετικά μπλοκ είναι εντελώς ανεξάρτητα μεταξύ τους. Αυτό σημαίνει ότι εάν τα δεδομένα μεταδίδονται μέσω δικτύου ή τηλεφωνικής γραμμής, σφάλματα μετάδοσης θα επηρεάσουν μόνο το μπλοκ που περιέχει το σφάλμα. Ωστόσο, το μπλοκ μπορεί να τροποποιηθεί, και η δράση αυτή θα περνάει απαρατήρητη. Πλεονέκτημά της μεθόδου αποτελεί το γεγονός ότι αν γίνει κάποιο λάθος σε κάποιο κομμάτι του κρυπτογραφημένου κειμένου, τότε αυτό επηρεάζει μόνο την αποκρυπτογράφηση του ίδιου και συνεπώς την εύρεση του συγκεκριμένου κομματιού του αρχικού κειμένου. Επιπρόσθετα, η παραλληλοποίηση της κρυπτογράφηση και αποκρυπτογράφησης διασφαλίζουν την ταχύτητα και ευκολία εφαρμογής, που την καθιστούν τη πιο κοινή λειτουργία του DES.

- CBC (Cipher Block Chaining): Στη μέθοδο αλυσιδωτής σύνδεσης τμημάτων κρυπτογραφίας (CBC) σε κάθε ομάδα κειμένου εφαρμόζεται η λογική συνάρτηση XOR (αποκλειστικό «ή») με δεύτερο μέλος το κρυπτογράφημα της προηγούμενης ομάδας και στη συνέχεια η έξοδος αυτού κρυπτογραφείται χρησιμοποιώντας το κλειδί. Σε αυτή τη λειτουργία, μπλοκ plaintext που είναι λιγότερο από 64 bits μήκος

μπορεί να είναι κρυπτογραφημένα. Κατά τη διαδικασία που επιτελείται, ένα 64-bit μπλοκ που ονομάζεται μητρώο Shift χρησιμοποιείται ως plaintext συμβολή DES, αφού έχει αρχικά οριστεί σε κάποια αυθαίρετη τιμή (και κρυπτογραφημένη μεσω του DES). ciphertext στη συνέχεια διέρχεται από ένα επιπλέον στοιχείο που ονομάζεται *M*-box, το οποίο απλά επιλέγει την άκρως αριστερή *M* bits του ciphertext, όπου *M* είναι ο αριθμός των bits στο μπλοκ που θέλουμε για την κρυπτογράφηση. Η τιμή αυτή γίνεται XOR με το πραγματικό απλό, και η έξοδος του ότι είναι ο τελικός ciphertext. Τέλος, το ciphertext ανατροφοδοτεί το Μητρώο Shift, και χρησιμοποιείται ως σπόρος plaintext για το επόμενο μπλοκ ώστε να είναι κρυπτογραφημένα. Μειονέκτημα του, είναι πως ακριβώς επειδή κάθε επόμενο κομμάτι κειμένου προς κρυπτογράφηση εξαρτάται από το προηγούμενο cipher, οποιοδήποτε σφάλμα μπορεί να προκύψει θα συμπαρασύρει αλληλουχία σφαλμάτων και στις ακόλουθες κρυπτογραφήσεις. Επιπρόσθετα, αυτός ο τρόπος λειτουργίας είναι πιο αργός απο τον ECB, το οποίο οφείλεται στην επιπρόσθετη πολυπλοκότητα.

- CFB (Cipher Feedback): Εδώ υπάρχει παρόμοια λειτουργία με την CBC. Με τον αλγόριθμο αυτό, το ίδιο κομμάτι κειμένου μπορεί να παράγει όμοιο cipher αν το αρχικό διάνυσμα είναι ίδιο και συνεπώς παράγεται το ίδιο κλειδί. Και σε αυτή τη λειτουγία διατηρείται το πρόβλημα της κατάρρευσης της διαδικασίας της αποκρυπτογράφησης σε περίπτωση αλλαγής στη διάταξη των ciphers.

- OFB (Output Feedback): Επίσης παρόμοια λειτουργία αποτελεί η OFB, με τη διαφορά όμως ότι η ποσότητα πληροφορίας στην οποία εφαρμόζεται η λογική πράξη XOR με την ομάδα καθαρού κειμένου δημιουργείται ανεξάρτητα από το καθαρό κείμενο ή το κρυπτογράφημα. Το βασικό πλεονέκτημα που παρουσιάζει η μέθοδος OFB σε σχέση με τη CFB είναι η διασφάλιση ότι σε περίπτωση εμφάνισης σφαλμάτων κατά τη μετάδοση, έχει περιορισμένη επίδραση και μόνο στα αντίστοιχα bits, άρα περιορίζεται η διάδοση σφαλμάτων. Ωστόσο,σε αντίθεση με τις υπόλοιπες τρεις, δεν θεωρείται εξίσου ασφαλής λόγω εύκολης τροποποίησης του αρχικού κειμένου[03],[14],[16].

AES

Το 1997, ο οργανισμός NIST ξεκίνησε την αναζήτηση του αντικαταστάτη του αλγορίθμου DES. Έτσι δημιουργήθηκε ο AES (Advanced Encryption Standard). Σε αντίθεση με τα προηγούμενα

κρυπτοσυστήματα, το AES δεν χρησιμοποιεί δίκτυο Feistel, κρυπτογραφεί ένα κείμενο με μέγεθος τμήματος 128 bits (τα διπλάσια από αυτά των προγόνων του), 194 ή 256 bits ενώ χρησιμοποιεί κλειδιά τριών μεγεθών, είτε 128, είτε 192 είτε 256 bits. Με σημείο αναφοράς το μέγεθος των κλειδιών προκύπτει και ο αριθμός των γύρων. Πιο συγκεκριμένα, για τον AES-128 έχουμε 10 γύρους, για τον AES-192 έχουμε 12 ενώ έχουμε 14 για τον AES-256. Ακόμα και στην πιο απλή εκδοχή (δηλαδή του AES-128) ο χώρος των πιθανών κλειδιών είναι $\mathbf{2^{128} \approx 3.4 \times 10^{34}}$, το οποίο καθιστά την εξαντλητική αναζήτηση αδύνατη καθώς θα χρειαζόταν πάροδος δισεκατομμυρίων χρόνων. Όσο για την αποκρυπτογράφηση του μηνύματος, είναι εφικτή με χρήση αναστροφής στον αλγόριθμο (φθίνουσα αρίθμηση των γύρων, ώστε να ισχύει η αντιστοιχία με τα κλειδιά του κάθε γύρου) και αντικατάστασης όλων των συναρτήσεων με τις αντίστροφές τους**.** Δηλαδή δεν βασίζεται στη δυσκολία εύρεσης του αλγορίθμου απο όσους διεξάγουν εξωτερική επίθεση, αλλά στην έλλειψη υπολογιστικής ισχύς που θα διαθέτουν.Ο AES έχει αποδειχθεί ανθεκτικός στις περισσότερες μορφές κρυπτανάλυσης που γνωρίζουμε ως τώρα, έχει απλούστερη μορφή σε σχέση με τον DES και έχει δημιουργηθεί με τρόπο κατάλληλο έτσι ώστε να υλοποιείται τόσο σε λογισμικό (software) όσο και σε hardware**.** [03,11,15]. Ο αλγόριθμος AES σχεδιάστηκε με τις ακόλουθες ιδιότητες:

- Αντοχή σε όλες τις μέχρι τότε γνωστές επιθέσεις

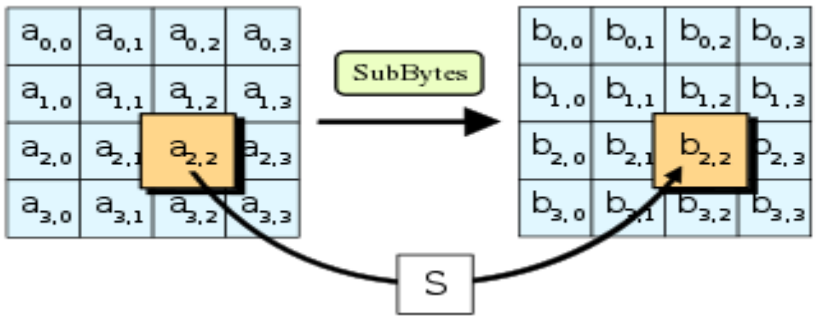
- ταχύτητα εκτέλεσης και οικονομία κώδικα κατά την υλοποίηση σε όλες τις διαθέσιμες πλατφόρμες

- απλότητα στη σχεδίαση.[15]

Ο αλγόριθμος αποτελείται από τα εξής τέσσερα στάδια:

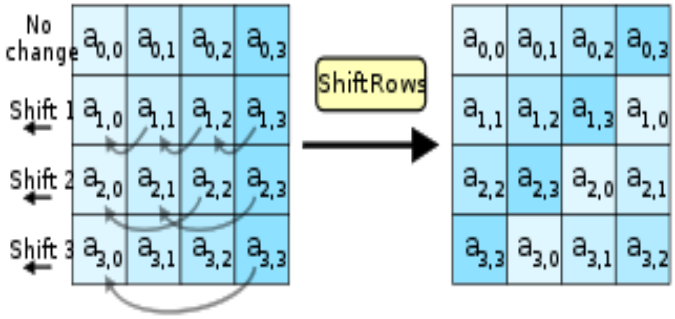
1) AddRoundKey: Αρχικά θα γίνει διαχωρισμός του κειμένου και του κλειδιού σε κελιά (ένα για κάθε χαρακτήρα) και δημιουργούμε δύο τετραγωνικούς πίνακες με στοιχεία τους χαρακτήρες των κελιών. Θα γίνει εφαρμογή της πράξης XOR (για να είναι εφικτή θα πρέπει όλοι οι χαρακτήρες να μετατραπούν σε δυαδικούς μεταξύ κάθε byte του κειμένου και του αντίστοιχου κελιού του κλειδιού. Συμπερασματικά θα έχουμε ως έξοδο έναν νέο πίνακα-block με στοιχεία τα αντίστοιχα αποτελέσματα της πράξης.

2) SubBytes: Ένα μη γραμμικό βήμα αντικατάστασης όπου κάθε byte αντικαθίσταται με ένα άλλο σύμφωνα με έναν πίνακα αναζήτησης. Δέχεται ως είσοδο τον πίνακα που εξήχθη κατά το προηγούμενο βήμα και μετατοπίζει κυκλικά τα bytes. Ο αριθμός της γραμμής εκφράζει το πρώτο ψηφίο του χαρακτήρα στο 16αδικό σύστημα, ενώ ο αριθμός της στήλης το δεύτερο ψηφίο του. Χρησιμεύει στη μετατροπή του αρχικού κειμένου σε cipher.



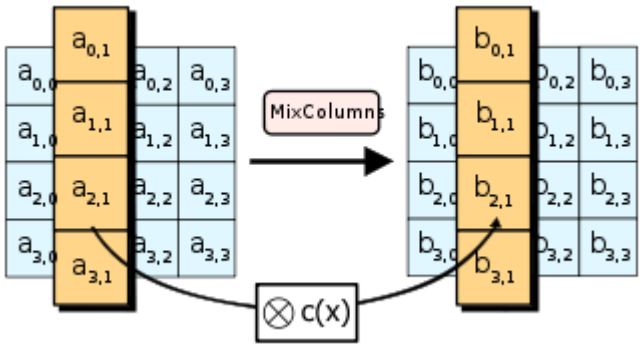
Εικόνα 2.4 : Στο βήμα SubBytes , κάθε byte στην κατάσταση αντικαθίσταται με την καταχώρισή του σε έναν σταθερό πίνακα αναζήτησης 8 bit

3) ShiftRows: Ένα βήμα μεταφοράς όπου οι τρεις τελευταίες σειρές της κατάστασης μετατοπίζονται κυκλικά σε έναν ορισμένο αριθμό βημάτων. Για το AES, η πρώτη σειρά παραμένει αμετάβλητη. Κάθε byte της δεύτερης σειράς μετατοπίζεται ένα προς τα αριστερά. Ομοίως, η τρίτη και η τέταρτη σειρά μετατοπίζονται με μετατοπίσεις δύο και τριών αντίστοιχα. Ως αποτέλεσμα, κάθε στήλη της κατάστασης εξόδου αποτελείται από bytes από κάθε στήλη της κατάστασης εισόδου. Η σημαντικότητα αυτού του βήματος έγκειται στην αποφυγή της ανεξάρτητης κρυπτογράφησης των στηλών, καθώς σε αυτή τη περίπτωση θα εκφυλιζόταν το AES σε τέσσερις ανεξάρτητους κωδικούς μπλοκ:



Εικόνα 2.5 : τα byte σε κάθε γραμμή της κατάστασης μετατοπίζονται κυκλικά προς τα αριστερά. Ο αριθμός των θέσεων που μετατοπίζεται κάθε byte διαφέρει σταδιακά για κάθε σειρά.

4) MixColumns: Μια λειτουργία γραμμικής μίξης που λειτουργεί στις στήλες της κατάστασης, συνδυάζοντας τα τέσσερα byte σε κάθε στήλη. Δηλαδή, τα τέσσερα byte κάθε στήλης της κατάστασης συνδυάζονται με χρήση αντιστρέψιμου γραμμικού μετασχηματισμού. Στη διάρκεια αυτής της λειτουργίας κάθε στήλη μετασχηματίζεται χρησιμοποιώντας έναν σταθερό πίνακα (ο οποίος θα πολλαπλασιάζεται κάθε φορά με μία στήλη και θα δίνει τη νέα τιμή της κατάστασης)[03,17]:



Εικόνα 2.6: κάθε στήλη της κατάστασης πολλαπλασιάζεται με ένα σταθερό πολυώνυμο *c(x).*

Συνοψίζοντας, τα πλεονεκτήματα των συμμετρικών συστημάτων είναι ότι απαιτείται σχετικά μικρός χρόνος κωδικοποίησης ή αποκωδικοποίησης, εύκολη μαθηματική υλοποίηση, ενώ η κρυπτανάλυση είναι δύσκολη. Ωστόσο, η ανάγκη για ασφαλή φύλαξη και διανομή του κλειδιού αποτελεί ένα μείζονος σημασίας μειονέκτημα[10].

2.4.2 Ασύμμετρη κρυπτογραφία

Με μια παρατήρηση της ιστορικής αναδρομής της κρυπτογραφίας, γίνεται εύκολα αντιληπτό ότι η μεγαλύτερη αδυναμία των περισσότερων κρυπτοσυστημάτων αποτελούσε ανέκαθεν η διανομή των κλειδιών. Ένας εισβολέας δύναται να υποκλέψει το κλειδί ανεξάρτητα του πόσο ισχυρό είναι το σύστημα, το οποίο πλέον δε θα είχε καμία χρησιμότητα. Οι κρυπτολόγοι είχαν πάντα ως σημείο αφετηρίας ότι το κλειδί της κρυπτογράφησης και της αποκρυπτογράφησης να είναι ταυτόσημα ή τουλάχιστον το ένα να προκύπτει εύκολα απ' το άλλο. Ωστόσο, η ανάγκη να μοιραστεί το κλειδί και σε άλλους χρήστες του συστήματος, το οποίο δημιούργησε το ζήτημα όχι μόνο της προστασίας του κλειδιού αλλά και της διαμοίρασής του με ασφαλή τρόπο. Η κρυπτογράφηση δημοσίου κλειδιού ή ασύμμετρου κλειδιού επινοήθηκε στο τέλος της

δεκαετίας του 1970, και η κύρια διαφορά με την συμμετρική κρυπτογράφηση είναι ότι εδώ έχουμε δύο κλειδιά, όπου τα δύο διαφορετικά κλειδιά που κατέχουν οι χρήστες έχουν διαφορετικές χρήσεις, το ένα χρησιμοποιείται για την κρυπτογράφηση και το άλλο για την αποκρυπτογράφηση. Πιο συγκεκριμένα, κάθε χρήστης διαθέτει δύο κλειδιά κρυπτογράφησης: το ένα ονομάζεται ιδιωτικό κλειδί και το άλλο δημόσιο κλειδί. Το ιδιωτικό κλειδί (κλειδί αποκρυπτογράφησης) θα πρέπει ο κάθε χρήστης να το προφυλάσσει και να το κρατάει κρυφό, ενώ αντιθέτως το δημόσιο κλειδί (κλειδί κρυπτογράφησης) θα πρέπει να το ανακοινώνει σε όλη την διαδικτυακή κοινότητα. Τα δύο αυτά κλειδιά έχουν μαθηματική σχέση μεταξύ τους. Αυτή βασίζεται κατά βάση στις μονόδρομες συναρτήσεις και πιο συγκεκριμένα, οι μονόδρομες συναρτήσεις «καταπακτής» (trapdoor one-way functions). Πρόκειται για αντιστρέψιμες συναρτήσεις οι οποίες είναι εύκολα υπολογίσιμες, ωστόσο ο υπολογισμός των αντιστρόφων τους είναι δύσκολος. Βασίζεται δηλαδή στην υπολογιστική πολυπλοκότητα που έχουν κάποια δύσκολα μαθηματικά προβλήματα όπως η παραγοντοποίηση ακεραίου σε πρώτους αριθμούς. Ακόμα και με γνώση της αντιστροφής διαδικασίας δεν θα υπήρχε η υπολογιστική ισχύ για την επιτυχημένη αποκρυπτογράφηση καθώς αυτή θα διαρκούσε εκατομμύρια χρόνια για έναν κοινό υπολογιστή[03,10,11,12,14].

Μαθηματικό υπόβαθρο για RSA

Με βάση τον ορισμό της διαιρετότητας , λέμε ότι ο a διαιρεί τον b (για $a, b \in \mathbb{Z}$) αν υπάρχει $c \in \mathbb{Z}$ τέτοιο ώστε να ισχύει $b = ac$. Σ' αυτή την περίπτωση ο a καλείται διαιρέτης του b και ο b πολλαπλάσιο του a. Επιπρόσθετα αν ο a διαιρεί τον b γράφουμε $a \mid b$. Αν ο a δεν διαιρεί τον b, τότε γράφουμε $a \nmid b$. Για παράδειγμα, $4|16$, $7|28$, $6 \nmid 10$. Μερικές βασικές ιδιότητες δίνονται στην παρακάτω:

Έχουμε $a, b, c \in \mathbb{Z}$. Τότε ισχύουν τα εξής:

(α) Αν $a|b$ και $b|c$, τότε $a|c$.

(β) Αν $a|b$ και $c|d$, τότε $ac|bd$.

(γ) Αν $a|b$ και $a|c$, τότε $a|bx + cy$, για κάθε $x, y \in \mathbb{Z}$.

(δ) Αν $a|b$ και $b \neq 0$, τότε $|a| \leq |b|$.

(ε) Αν a|b και b|a, τότε |a| = |b|.

Με βάση την ευκλείδια διαίρεση, αν a, b θετικοί ακέραιοι, τότε υπάρχουν μοναδικά q, r ∈ ℤ τέτοια ώστε a = bq + r με 0 ≤ r < b. Ο ευκλείδιος αλγόριθμος είναι μια διαδικασία η οποία δέχεται ως είσοδο δύο ακέραιους αριθμούς και όταν ολοκληρωθεί δίνει τον μέγιστο κοινό τους διαιρέτη. Για a, b θετικοί ακέραιοι με a ≥ b. Ο αλγόριθμος υπολογίζει τον gcd(a, b) στηριζόμενος στην σχέση a = bq + r ⇒ gcd(a, b) = gcd(b, r):

i. Θέτω r_0 = a, r_1 = b και j= 1.

ii. Διαιρώ το r_{j-1} με το r_j και παίρνω $r_{j-1} = r_j q_j + r_{j+1}$.

iii. Αν r_{j+1}= 0 τότε gcd(a, b) = r_j , διαφορετικά θέτω j = j + 1 και γυρνάω στο προηγούμενο βήμα.

Ο μέγιστος κοινός διαιρέτης των a, b είναι ο μεγαλύτερος ακέραιος που διαιρεί τον a και τον b. συμβολίζουμε με (a, b), ή με ΜΚΔ(a, b), ή με gcd(a, b). Πρώτος αριθμός λέγεται ένας ακέραιος μεγαλύτερος του 1 που δεν έχει άλλους διαιρέτες εκτός από το 1 και τον εαυτό του. Ένας αριθμός που δεν είναι πρώτος λέγεται σύνθετος. Κάθε ακέραιος μεγαλύτερος του 1 είναι είτε πρώτος είτε γινόμενο πρώτων αριθμών, ενώ σύμφωνα με το θεώρημα του Ευκλείδη, οι πρώτοι είναι άπειροι σε πλήθος. Σχετικώς πρώτοι (coprime) ονομάζονται οι αριθμοί a, b αν ισχύει ότι ΜΚΔ(a,b)=1. Επιπρόσθετα,ιδιαίτερα σημαντικό είναι το θεώρημα κατά το οποίο η ανάλυση ενός ακέραιου αριθμού σε γινόμενο πρώτων παραγόντων είναι μονοσήμαντη αν δεν ληφθεί υπόψη η σειρά των παραγόντων.

Η modular αριθμητική αποτελεί ένα σύστημα αριθμητική ακεραίων, κατά το οποίο οι αριθμοί περιτυλίσσονται γύρω από μία τιμή, το modulo (γράφεται σύντομα "mod") ή υπόλοιπο. Δηλαδή αν θέλουμε να υπολογίσουμε με τι ισούται ένας συγκεκριμένος αριθμός *m* modulo έναν αριθμό *n*, εκτελούμε ακέραια διαίρεση και το αποτέλεσμά μας είναι το υπόλοιπο της διαίρεσης. Π.χ. 27 ≡ 2 (mod 5). Η σχέση " ≡ mod n" ονομάζεται «ισοτιμία». Εαν a · b ≡ 1 (mod n), ο b είναι αντίστροφος για το α (mod n). Ο Ευκλείδιος αλγόριθμος που ορίσαμε προηγουμένως, μπορεί να ελέγξει αν οι δύο αριθμοί έχουν κοινούς παράγοντες. Η αριθμητική modular καλείται και ωρολογιακή αριθμητική, καθώς ασχολούμαστε με ομάδα αριθμών με κυκλική διάταξη, όπως ακριβώς οι αριθμοί σε ένα ρολόι (το οποίο αποτελεί παράδειγμα εργασίας με modulo 12).

Έστω ένα μη κενό σύνολο X με $a,b \in X$ τότε γράφουμε $a \sim b$. Μία σχέση $X \subseteq A \times A$ ονομάζεται σχέση ισοδυναμίας αν ισχύουν :

1. $a \sim a, \forall\, a \in A$
2. $a \sim b \Rightarrow b \sim a$
3. $a \sim b$ και $b \sim c \Rightarrow a \sim c$

Έχοντας λοιπόν μία σχέση ισοδυναμιάς επί του Α ορίζουμε ως κλάση ισοδυναμίας το σύνολο $[a] = \{x\, \varepsilon\, A | x \sim a\}$.

Η έννοια της σχέσης ισοδυναμίας ήταν απαραίτητη για να οριστεί το σύνολο $\mathbb{Z}_m$. Αν έχουμε $m > 0$ έναν φυσικο αριθμό και θεωρήσουμε τη σχέση ισοδυναμίας $a \sim b \Leftrightarrow a \equiv b\ mod\ m$, τότε έχουμε τη κλάση ισοδυναμίας του a, $[a] = \{x\, \varepsilon\, \mathbb{Z} \mid x \sim a\} = \{x\, \varepsilon\, \mathbb{Z} \mid x \equiv a\ mod\ m\} = \{a + km | \ k\, \varepsilon\, \mathbb{Z}\}$.Το σύνολο των κλάσεων ισοδυναμίας του *a* καλείται σύνολο των ακεραίων *modulo m* και συμβολικά το γράφουμε $\mathbb{Z}_m$. Για *a,b* στοιχεία του $\mathbb{Z}_m$, μπορούμε να ορίσουμε τη πρόσθεση ως $[a] + [b] = [a + b]$ και τον πολλαπλασιασμό ως $[a][b] = [ab]$. Αν έχουμε ένα στοιχείο ā συμμετρικό του *a,* τότε ισχύει $\bar{a} + a = 0 = m$.

Αν έχουμε $\gcd(a, m) = 1$, τότε το *a* είναι αντιστρέψιμο στο $\mathbb{Z}_m$ με το *b* να αποτελεί αντίστροφο στοιχείο και ισχύει $ab \equiv 1\ mod\ m \Leftrightarrow ab = 1$. Το σύνολο των αντιστρέψιμων στοιχείων συμβολίζεται ως $U(\mathbb{Z}_m) = \{\bar{a} \in \mathbb{Z}_m, \gcd(a, m) = 1\}$. Το πλήθος των στοιχείων αυτού του συνόλου ορίζει την συνάρτηση Euler *φ(m)*. Δηλαδή το πλήθος των ακεραίν που είναι σχετικώς πρώτοι με το *m.* Δηλαδή ορίζεται ως: $\varphi(m) = |\{1 \leq a < m : \gcd(a, m)\}| = 1$. Π.χ. $\varphi(3) = 2, \varphi(81) = 54$. Επιπρόσθετα, αν ο p αναλυθεί σε γινόμενο πρώτων p_1 και p_2 τότε συνεπάγεται πως : $\varphi(p) = \varphi(\, p_1\, p_2\,) = \varphi(\, p_1)\varphi(\, p_2\,) = (p_1 - 1)(p_2 - 1)$, καθώς αν ο p είναι πρώτος τότε ισχύει $\varphi(p) = p - 1$.

Γνωστό ως θεώρημα των Fermat-Euler αποτελεί το εξής: Ας είναι $n, a \in \mathbb{Z}, \mathrm{n} > 1$ και $\gcd(a, n) = 1$. Τότε ισχύει $a^{\varphi(n)} \equiv 1 (mod\ n)$.

Εξίσου σημαντικό το μικρό θεώρημα του Fermat κατά το οποίο αν έχουμε *p* πρώτο και *a* ακέραιο με $p \nmid \mathrm{a}$, τότε $a^{p-1} \equiv 1 (mod\ p)$. Μέσω αυτού, το θεώρημα των Fermat-Euler δίνει την αποδεικτέα σχέση $a^p \equiv a (mod\ p)$.

Ένα χρήσιμο πόρισμα που εξάγεται σχετικά με τη συνάρτηση Euler αποτελεί το γεγονός ότι αν έχουμε έναν ακέραιο n ο οποίος είναι μεγαλύτερος του 2, τότε ο ακέραιος $\varphi(n)$ είναι άρτιος.[01-05]

RSA

Ο αλγόριθμος RSA είναι ένας από τους πρώτους αποτελεσματικούς αλγόριθμους δημοσίου κλειδιού που δημοσιεύτηκε το 1976, και είναι ένα από τα πιο διαδεδομένα κρυπτοσυστήματα δημοσίου κλειδιού. Τα μαθηματικά που βρίσκονται στη βάση του αλγορίθμου RSA αποτελούν εκμετάλλευση στοιχειωδών αρχών της θεωρίας αριθμών και ιδιαίτερα των πρώτων αριθμών. Ακόμα πιο συγκεκριμένα, η ασφάλειά του βασίζεται στο πρόβλημα της παραγοντοποίησης ενός σύνθετου ακεραίου σε γινόμενο πρώτων παραγόντων. Επιπρόσθετα, οσο μεγαλύτερος είναι ο ακέραιος που παραγοντοποιείται, τόσο πιο δύσκολο είναι να βρεθούν οι πρώτοι παράγοντες. Εφόσον δεν υπάρχει ταχύς αλγόριθμος ο οποίος να εντοπίζει τους πρώτους παράγοντες ενός ακεραίου, συνεχίζει να αποτελεί ένα ασφαλές σύστημα, για αυτό είναι και ευρέως χρησιμοποιούμενο[03,10,12,18]. Ο RSA για την δημιουργία του ζεύγους κλειδιών (δημόσιο και ιδιωτικό κλειδί) και την κρυπτογράφηση-αποκρυπτογράφηση εκτελεί τα παρακάτω βήματα:

1. Επιλέγονται δύο μεγάλοι πρώτοι αριθμοί (p,q).

2. Πραγματοποιείται υπολογισμός του $n = p * q$.

3. Υπολογίζεται το $\varphi(n) = (p-1) * (q-1)$.

4. Επιλέγεται ακέραιος e που να βρίσκεται στο διάστημα $(1, \varphi(n))$ και $\gcd(e, \varphi(n)) = 1$.

5. Υπολογίζεται το d ώστε $e * d = 1 mod \varphi(n)$.

6. Τέλος δημιουργείται το δημόσιο και το ιδιωτικό κλειδί όπου δημόσιο το $\{n, e\}$ και ιδιωτικό το $\{n, d\}$.

7. Σχετικά με την κρυπτογράφηση των μηνυμάτων χρησιμοποιείται η συνάρτηση $E(M)$ όπου M το μήνυμα και η συνάρτηση E ορίζεται ως: $E(M) = M^e mod\ n$. n. Όσο για την αποκρυπτογράφηση των μηνυμάτων χρησιμοποιείται η συνάρτηση $D(E(M))$ όπου

$E(M)$ το κρυπτογραφημένο μήνυμα και η συνάρτηση D ορίζεται ως : $D(E(M)) = E(M)^{d} mod\ n.$ [11]

Η φ αποτελεί τη συνάρτηση του Euler η οποία επί της ουσίας δείχνει το πλήθος των θετικών ακέραιων αριθμών που είναι μικρότεροι από n και πρώτοι με αυτόν, ενώ η d ονομάζεται πολλαπλασιαστική ανάστροφος του αριθμού e [14] . Παρατηρούμε ότι η ασφάλεια του αλγορίθμου έγκειται στη δυσκολία εύρεσης των p και q, δεδομένου του e και του n. Ο κρυπταναλυτής πρέπει να εντοπίσει τους αριθμούς p και q ώστε να παράξει το φ και στη συνέχεια από το γνωστό e να υπολογίσει το d. Η διαδικασία εντοπισμού των p και q, η παραγοντοποίηση (factorization) δηλαδή του n, ή αλλιώς του modulus είναι ως σήμερα αδύνατη για τιμή μήκους από 1024bit και πάνω. Για το λόγο αυτό, σε πραγματικά κρυπτοσυστήματα χρησιμοποιούνται ιδιαίτερα μεγάλοι πρώτοι αριθμοί [15]. Ακολουθεί ένα παράδειγμα χρήσης του RSA με χρησιμοποίηση μικρών πρώτων αριθμών για ευκολία στις πράξεις και καλύτερη κατανόηση του αλγορίθμου:

- Εντοπίζουμε δύο πρώτους αριθμούς για να δημιουργήσουμε το ζεύγος κλειδιών. Έστω $p = 11$ και $q = 5$.
- Από αυτούς θα προκύψει $n = p * q = 11 * 5 = 55$.
- Υπολογισμός $\varphi(n)$ για $n = 55$ και έτσι έχουμε $\varphi(55) = (11 - 1) * (5 - 1) = 10 * 4 = 40$.
- Εντοπίζουμε e τέτοιο ώστε να είναι σχετικά πρώτος με το 40. Έστω $e = 3$.
- Το δημόσιο κλειδί επομένως είναι $\{3,55\}$.
- Υπολογίζουμε το ιδιωτικό κλειδί d, τέτοιο ώστε $ed\ mod\ \varphi(n) = 1$. Ένας τέτοιος αριθμός είναι το 27. Άρα το ιδιωτικό κλειδί είναι το $\{27,55\}$.

Έστω ότι κάποιος θέλει να κρυπτογραφήσει τον αριθμό 7 με τη χρήση του RSA και να μας στείλει το κρυπτογράφημα. Για την κρυπτογράφηση θα χρησιμοποιήσει το διαθέσιμο δημόσιο κλειδί και θα υπολογίσει:

$$c = m^{e} mod\ n = 7^{3} mod\ 55 = 13$$

Άρα θα μας αποστείλει τον αριθμό 313. Για να τον αποκρυπτογραφήσουμε, θα χρησιμοποιήσουμε το ιδιωτικό κλειδί, που μόνο εμείς κατέχουμε. Άρα έχουμε [15]:

$$m = c^d mod\ n = 13^{27} mod 55 = 7$$

Για την ώρα ο μόνος δρόμος για να παραβιάσει κανείς την ασφάλεια που παρέχει ο αλγόριθμος RSA είναι να επιλυθεί το πρόβλημα της παραγοντοποίησης και να βρει τους παράγοντες του *n*. Αυτό σημαίνει πως θα πρέπει να υπάρχει κάποιος ισχυρός υπολογιστής που να υπολογίζει γρήγορα όλους τους πιθανούς παράγοντες για μεγάλες τιμές του *n* ή πως θα ανακαλυφθεί κάποια καινούργια μαθηματική μέθοδος υπολογισμού των παραγόντων περισσότερο αποτελεσματική. Για τις σημερινές εφαρμογές ένα κλειδί μεγέθους 2048 θεωρείται ισχυρό[12].

Diffie-Hellmann

Ο πρώτος αλγόριθμος για ασύμμετρο κρυπτογραφικό σύστημα δημοσιεύτηκε το 1976 στην εργασία των Diffie – Hellman που όριζε την κρυπτογραφία με ασύμμετρο κρυπτογραφικό σύστημα και είναι γνωστός ως ανταλλαγή κλειδιών κατά Diffie – Hellman. Δεν αποτελεί ολοκληρωμένο κρυπτοσύστημα, διότι δεν στοχεύει στην ανταλλαγή πολλών πληροφοριών μεταξύ αποστολέα και δέκτη. Στόχος του είναι η εφικτή και ασφαλή ανταλλαγή, μεταξύ δυο χρηστών, ενός μυστικού κλειδιού, το οποίο στη συνέχεια θα χρησιμοποιηθεί για κρυπτογράφηση μηνυμάτων. Αυτό το πρωτόκολλο βασίζεται στη δυσκολία πρόβλημα του διακριτού λογαρίθμου, ο οποίος ορίζεται ως εξής:

Αρχικά προσδιορίζεται μία πρωτογενής ρίζα *a* ενός πρώτου αριθμού *p*, του οποίου οι δυνάμεις παράγουν όλους τους ακεραίους από το 0 έως το *p-1*. Συνεπώς αν *a* μία ρίζα του πρώτου αριθμού *p*, τότε οι αριθμοί $a\ mod\ p, a^2\ mod\ p, \dots, a^{p-1}\ mod\ p$ αποτελούν τους ακεραίους από *1* ως το *p-1* με κάποια μετάθεση. Για οποιοδήποτε ακέραιο b και για μία πρωτογενή ρίζα a ενός πρώτου αριθμού *p*, υπάρχει μοναδικός πρώτος αριθμός *i*, τέτοιος ώστε $b \equiv a^i\ mod\ p$ με $0 \leq i \leq p-1$. Ο εκθέτης *i* είναι ο διακριτός λογάριθμος του *b* για τη βάση $a\ mod\ p$.[14]

Με βάση αυτό η διαδικασία του Diffie-Hellmann για ανταλλαγή κλειδιών έχει ως εξής:

1. Αρχικά επιλέγεται μεγάλος πρώτος αριθμός *p* και ένας γεννήτορας $g \in \mathbb{Z}_p$.

2. Ο αποστολέας επιλέγει έναν τυχαίο $a \in \mathbb{Z}_p$ που τον γνωρίζει μόνο αυτός και στέλνει στον παραλήπτη το μήνυμα $y_a \equiv g^a mod\ p$.

3. Ο παραλήπτης επιλέγει έναν τυχαίο $b \in \mathbb{Z}_p$ που τον γνωρίζει μόνο αυτός και στέλνει στον παραλήπτη το μήνυμα $y_b \equiv g^b mod\ p$.

4. Ο παραλήπτης λαμβάνει το $g^a mod\ p$ και υπολογίζει το $K \equiv (g^a)^b mod\ p$.

5. Ο αποστολέας λαμβάνει το $g^b mod\ p$ και υπολογίζει το $K \equiv (g^b)^a mod\ p$.

6. Συμπερασματικά, αποστολέας και παραλήπτης συμφώνησαν στο κοινό κλειδί *K*. [18]

Ακολουθεί απλό παράδειγμα εφαρμογής του Diffie-Hellmann για τη δημιουργία κοινού κλειδιού *d*. Αρχικά επιλέγουμε $p = 71$ και $g = 53 \in \mathbb{Z}_{71}$. Με τυχαίο τρόπο ο αποστολέας επιλέγει $a = 13$ και ο παραλήπτης $b = 21$. Ο αποστολέας υπολογίζει $A \equiv 53^{13} mod\ 71 \equiv 56$. Αντίστοιχα για τον παραλήπτη έχουμε $B \equiv 53^{21} mod\ 71 \equiv 66$. Έπειτα, πραγματοποιείται ανταλλαγή κλειδιών A και B με τον πρώτο να υπολογίζει $B^a \equiv 66^{13} mod\ 71 \equiv 17$. Αντίστοιχα, ο παραληπτης υπολογίζει $A^b \equiv 56^{21} mod\ 71 \equiv 17$. Και με αυτόν τον τρόπο έχει δημιουργηθεί το *17* ως το κοινό τους μυστικό κλειδί[03].

Το Diffie-Hellmann χρησιμοποιείται για την ασφάλεια ποικίλων υπηρεσιών διαδικτύου. Ωστόσο, έρευνα που δημοσιεύτηκε τον Οκτώβριο του 2015 υποδηλώνει ότι οι παράμετροι που χρησιμοποιούνται για πολλές εφαρμογές διαδικτύου, δεν είναι αρκετά ισχυροί ώστε να αποφευχθεί ο συμβιβασμός από πολύ καλά χρηματοδομούμενους επιτιθέμενους, όπως οι υπηρεσίες ασφάλειας ορισμένων χωρών[43].

Κεφάλαιο 3
Κβαντική Φυσική

Προκειμένου να οριστεί το πεδίο της κβαντικής κρυπτογραφίας πρέπει να αποσαφηνιστούν πολλές έννοιες της κβαντικής θεωρίας. Με ιδέες από αυτό το γνωστικό πεδίο και με τον κατάλληλο συνδυασμό με τη κλασική κρυπτογραφία πραγματοποιήθηκε η δόμησή της.

3.1 Εισαγωγή στη κβαντική θεωρία

Η κβαντομηχανική (επίσης γνωστή ως κβαντική μηχανική ή κβαντική φυσική) είναι μια θεωρία της φυσικής μηχανικής. Θεωρείται πιο θεμελιώδης από την κλασική μηχανική, καθώς εξηγεί φαινόμενα που η κλασική μηχανική και η κλασική ηλεκτροδυναμική αδυνατούν να αναλύσουν. Δύο χρονικές περίοδοι διαιρούν τη κβαντική μηχανική: Από τα τελη του 19ου αιώνα μέχρι το 1923 (γνωστή ως περίοδος της παλαιάς κβαντικής θεωρίας), και πέραν του 1923 όπου γίνεται έντονη σύνδεση της κβαντομηχανικής με την επιστήμη των υπολογιστών. Ο όρος «κβάντο» (quantum)

φέρει τις ρίζες του από την αντίστοιχη λατινική λέξη που σημαίνει «ποσό», και αναφέρεται σε ποσότητες φυσικού μεγέθους οι οποίες παίρνουν διακριτές τιμές. Η κβαντομηχανική με το πέρας ενός αιώνα πειραματισμού δεν έχει διαψευστεί. Κρύβεται πίσω από πολλά φυσικά φαινόμενα και ιδιαιτέρως τα χημικά φαινόμενα καθώς και τη φυσική της στερεάς κατάστασης. Κεντρικό ρόλο στη δημιουργία της διαδραμάτισε η απόδειξη του Maxwell ότι το φως είναι εγκάρσια ηλεκτρομαγνητικά κύματα. Έπειτα ο Einstein διατύπωσε ότι το φως διαδίδεται σε μικρά πακέτα ενέργειας που καλούνται φωτόνια ή κβάντα. Η ενέργεια των σωματιδίων αυτών είναι ανάλογη με την συχνότητα επί τη σταθερά του Planck ($\boldsymbol{h = 6.626 \cdot 10^{-34} J \cdot sec}$), δηλαδή είναι ίση με $\boldsymbol{E = h \cdot f}$. [19,20]

Τα φωτόνια διαδίδονται με τη ταχύτητα του φωτός ($\boldsymbol{c = 3 \cdot 10^8 \frac{m}{s}}$) και έχουν μάζα ηρεμίας ίση με μηδέν. Η συχνότητά τους f θα είναι ανάλογη του μήκους κύματος λ καθώς: $\boldsymbol{\lambda = \frac{c}{f} \Leftrightarrow f = \frac{c}{\lambda}}$. Συνεπώς η ενέργεια των φωτονίων θα είναι: $\boldsymbol{E = h \cdot f = h\frac{c}{\lambda}}$. Αντίστοιχα και η ορμή θα είναι ανάλογη της ενέργειας:

$$\boldsymbol{p = \frac{E}{c} = \frac{\frac{hc}{\lambda}}{c} = \frac{h}{\lambda}}$$

Μάλιστα, αν εισάγουμε ένα κυματικό αριθμό ίσο με $\boldsymbol{k = \frac{2\pi}{\lambda}}$, και με γνωστό ότι η ανηγμένη σταθερά του Planck είναι $\hbar = \frac{h}{2\pi}$ τότε η ορμή γίνεται:

$$\boldsymbol{p = \frac{h}{\lambda} = \frac{\hbar\, 2\pi}{\frac{2\pi}{k}}} = \hbar$$

Η κβαντική ηλεκτροδυναμική τις πρώτες δεκαετίες του εικοστού αιώνα, επικύρωσε πλήρως τη διττή φύση του φωτός, δηλαδή τόσο τον κυματικό όσο και τον σωματιδιακό του χαρακτήρα. Ο δεύτερος εκδηλώνεται σε φαινόμενα αλληλεπίδρασης της ύλης με το φως, ενώ ο πρώτος σε φαινόμενα περίθλασης και συμβολής. Η ενέργεια ενός φωτονίου στην ειδική θεωρία σχετικότητας είναι (Δεδομένης της μηδενικής μάζας ηρεμίας κάθε φωτονίου):

$$\boldsymbol{E^2 = (pc)^2 + (mc^2)^2 \overset{m=0}{\Longrightarrow} E = pc \Rightarrow p = \frac{E}{c}}$$

Συνεπώς, κάθε ηλεκτρομαγνητικό κύμα έχει κβαντωμένη ενέργεια δηλαδή υλοποιείται σε δέσμες (κβάντα) φωτός ή φωτόνια. Τα κυματοσωματιδιακά χαρακτηριστικά κάθε φωτονίου δίνονται από τις σχέσεις $\boldsymbol{E = hf}$ και $\boldsymbol{E = pc}$.

Πριν τον Einstein, ο Max Planck διατύπωσε πως το φως εκπέμπεται και απορροφάται από τα άτομα της ύλης «ασυνεχώς», και κάθε σύστημα έχει ενέργεια ίση με $\boldsymbol{E = nhf = nh\frac{\omega}{2\pi} = n\,\hbar\omega}$, όπου ω η κυκλική συχνότητα και $\boldsymbol{n = 1, 2, 3, \ldots}$ ο κβαντικός αριθμός. Δηλαδή το φως δε μπορεί να πάρει οποιαδήποτε τιμή ενέργειας λόγω της κυκλικής του συχνότητας και της περιοδικής του μεταβολής. Συνεπώς, από το άτομο δεν εκπέμπονται κύματα με συνεχή τρόπο, αλλά φωτόνια που χαρακτηρίζονται από μία συχνότητα και μία ορισμένη ποσότητα ενέργειας (που θα έχει διακριτή τιμή).

Εκτός από τον σωματιδιακό χαρακτήρα του φωτός που καταλήξαμε με σαφή τρόπο προηγουμένως, ο κυματικός χαρακτήρας της κίνησης της ύλης διατυπώθηκε απο τον De Broglie. Διατύπωσε τη κίνηση των φωτονίων ως σωματίδια και υλικά κύματα με ορμή:

$$\boldsymbol{p = \frac{E}{c} = \frac{\frac{hc}{\lambda}}{c} = \frac{h}{\lambda}}$$

Επίσης έχουν συχνότητα f που δίνεται από τον τύπο[20]:

$$\boldsymbol{E = hf \Leftrightarrow f = \frac{E}{h}}$$

3.2 Αρχή της αβεβαιότητας

Η δυϊκή φύση του φωτός (κύμα-σωματίδιο) που ήταν αποτέλεσμα τόσο των φαινομένων συμβολής όσο και αυτών της κβάντωσης της ενέργειας συσχετίστηκε σύντομα με την αβεβαιότητα (ή απροσδιοριστία) από τον Heisenberg. Τη διατύπωσε έπειτα από στενή συνεργασία με τον Bohr και έδινε μία ολοκληρωτικά νέα ερμηνεία για τον φυσικό κόσμο όπως ότι κύμα και σωματίδιο αποτελούν διαφορετικές θεωρήσεις του ίδιου πράγματος. Αυτή ήταν η εκκίνηση της αντικατάστασης του ντετερμινισμού της κλασικής φυσικής με την τυχαιότητα των γεγονότων. Πιο συγκεκριμένα, στη κλασική μηχανική, υπολογίζονται οι χωρικές συντεταγμένες κάθε σημειακού σωματιδίου με μεγάλη ακρίβεια οποιαδήποτε χρονική στιγμή. Αντίθετα, στη κβαντική μηχανική, δε μπορεί να υπολογιστεί ταυτοχρόνως η ορμή και η θέση ενός σωματιδίου. Η

μαθηματική διατύπωση αυτής της αρχής περιγράφει ότι το γινόμενο των αβεβαιοτήτων θέσης ($\boldsymbol{\Delta\chi}$) και ορμής ($\boldsymbol{\Delta p}$) ενός σωματιδίου, δε μπορεί να είναι μικρότερο από το μισό της σταθεράς του Planck:

$$\boldsymbol{\Delta\chi\ \Delta p} > \frac{\hbar}{\mathbf{2}}$$

Αφού τα σωματίδια συμπεριφέρονται ως κύμα (όπως και κάθε κύμα) θα πρέπει να περιγράφονται από μία κυματοσυνάρτηση $\boldsymbol{\Psi(x,t)}$, που εξαρτάται από τη θέση και τον χρόνο όλων των σωματιδίων του συστήματος που περιγράφει. Παρά το γεγονός ότι μία κυματοσυνάρτηση περιγράφει μία μεταβολή κάποιας φυσικής ποσότητας, στη περίπτωση ενός σωματιδίου θα αδυνατούσε να περιγραφεί. Εδώ η κυματοσυνάρτηση εκφράζει πιθανότητα. Δίνει δηλαδή το πλάτος πιθανότητας να βρεθεί το σωμάτιο στη μία ή στην άλλη περιοχή του χώρου[13,18,20].

3.3 Κβαντική Διεμπλοκή

Η βασική ιδέα για την έννοια της κβαντικής διεμπλοκης (ή σύζευξης) ήταν η απόδειξη ότι η κβαντομηχανική ήταν ελλιπής, καθώς δεν υπήρχαν ορισμένες παράμετροι που ονομάστηκαν «κρυμμένες μεταβλητές». Για να επιτευχθεί αυτό χρησιμοποιήθηκαν δύο κβαντικά συστήματα, τα οποία αφού αλληλεπίδρασαν απομακρύνθηκαν μεταξύ τους. Ωστόσο, με άγνωστο τρόπο συνέχισαν να αλληλεπιδρούν μεταξύ τους, και ως αποτέλεσμα, οι φυσικές μετρήσεις των δύο στοιχείων επηρεάζουν η μία την άλλη. Ο όρος διεμπλοκή χρησιμοποιήθηκε πρώτη φορά από τον Schrodinger το 1935 για να περιγράψει την αλληλεπίδραση των απομακρυσμένων κβαντικών συστημάτων. Συνοδεύτηκε από το γνωστό του νοητικό πείραμα με βάση το οποίο μία γάτα μπορεί να είναι ταυτόχρονα ζωντανή και νεκρή, καθώς και απο τη δημιουργία του όρου κβαντική υπέρθεση, που είναι ιδιαίτερα θεμελιώδης στη κβαντική φυσική. Σύμφωνα με τη κβαντική υπέρθεση (ή αρχή της επαλληλίας) για κάθε γραμμικό σύστημα το ολικό αποτέλεσμα ενός φαινομένου που αποτελείται από επί μέρους φαινόμενα, είναι ίσο με το άθροισμα των επιμέρους αποτελεσμάτων. Έτσι, κάθε κβαντικό σύστημα μπορεί να βρεθεί σε κατάσταση υπερθέσεως δύο βασικών καταστάσεων, και θα αποδίδεται από τη μαθηματική έκφραση που εμπερικλείει τα πλάτη πιθανότητας. Από την έννοια της υπέρθεσης καθίσταται σαφές πως δεν υπάρχει δυνατότητα μέτρησης της κατάστασης ενός κβαντικού συστήματος που βρίσκεται σε υπέρθεση δύο καταστάσεων, αλλά μπορεί να υπολογιστεί με σαφήνεια μόνο μία εξ αυτών. Αφού αποσαφηνίστηκε η σημαντική ερμηνεία της υπέρθεσης, επιστρέφουμε στο πείραμα κατά το

οποίο μία γάτα κλείνεται σε ένα θάλαμο με μέγιστη διάρκεια μίας ώρας μαζί με μία διάταξη. Η τελευταία είναι ένα κβαντικό σύστημα που βρίσκεται σε επαλληλία καταστάσεων εκπομπής και μη εκπομπής εντός εμβέλειας κίνησης της γάτας. Υπάρχει ποσότητα ραδιενεργής ουσίας ικανής να διασπάσει ή όχι ένα από τα άτομά της με ίσες πιθανότητες (εντός του χρονικού ορίου της μίας ώρας). Με την ενεργοποίηση του μετρητή ελευθερώνεται υδροκυάνιο από τη φιάλη, και έτσι όταν ανοίξουμε τον θάλαμο η γάτα θα είναι μόνο ζωντανή ή νεκρή (όχι όμως και τα δύο). Έτσι γεννιέται το ερώτημα «πότε παύει ένα κβαντικό σύστημα να βρίσκεται σε κβαντική υπέρθεση καταστάσεων και να γίνεται ή το ένα ή το άλλο;». Συνεπώς, ένας ορισμός της κβαντικής διεμπλοκής είναι ότι αποτελεί το φαινόμενο κατά το οποίο δύο σωματίδια που αλληλεπιδρούν συνενώνοντας τις κυματοσυναρτήσεις τους και μένουν σε κατάσταση διεμπλοκής μεταξύ τους, ασχέτως του χώρου που μεσολαβεί από το ένα στο άλλο. Ενώ μία πιο φορμαλιστική προσέγγιση είναι ότι δύο κβαντικά συστήματα βρίσκονται σε κβαντική διεμπλοκή όταν η κατάστασή τους δε μπορεί να γραφτεί ως τανυστικό γινόμενο των βασικών τους καταστάσεων[18,20,22].

3.4 Παράδοξο Einstein- Podolsky - Rozen (EPR)

Άμεση απόρροια του παράδοξου της κβαντικής διεμπλοκής είναι το EPR παράδοξο που παρουσιάστηκε το 1935 και αμφισβήτησε τη πληρότητα της εξηγησης της κβαντομηχανικής. Δηλαδή ότι τα φυσικά συστήματα δεν έχουν καθορισμένες ιδιότητες μέχρι τη μέτρησή τους, η οποία προκαλεί κατάρρευση της κυματοσυνάρτησης. Όπως και ότι τα αποτελέσματα που θα εξαχθούν χρησιμεύουν στη πρόβλεψη μόνο των πιθανοτήτων κατανομής μιας μέτρησης. Αποτελεί ένα νοητικό πείραμα στο οποίο πρέπει πρώταν να εισαχθεί η έννοια του spin να γίνει κατανοητό. Το spin αποτελεί μία ιδιότητα η οποια ανήκει εγγενώς σε όλα τα στοιχειώδη σωματίδια. Σημαίνει ότι τα σωματίδια έχουν στροφορμή και προσανατολισμό στο χώρο και όχι ότι πραγματικά περιστρέφονται. Πρέπει να γίνει επιλογή της διεύθυνσης ως προς την οποία μπορεί να μετρηθεί το spin ενός σωματιδίου. Τα αποτελέσματα που μπορεί να δώσει η μέτρηση είναι δύο:

- Είτε θα ειναι πάνω ($\boldsymbol{spin} = \frac{\hbar}{2}$) δηλαδή ευθυγραμμισμένο με τη διεύθυνση μέτρησης

- Είτε θα ειναι κάτω ($\boldsymbol{spin} = -\frac{\hbar}{2}$) δηλαδή αντίθετο με τη διεύθυνση μέτρησης

Όπως αντίστοιχα για το ηλεκτρόνιο οι τιμές για το spin είναι $\frac{1}{2}$ και -$\frac{1}{2}$. Ωστόσο, σύμφωνα με την αρχή της απροσδιοριστίας δεν είμαστε σε θέση να γνωρίζουμε τη τιμή του spin πριν τη μέτρησή του. Μπορεί να είναι ένα μίγμα δυνατών τιμών, αλλά θα μπορεί κανείς να γνωρίζει μόνο μετά τη διαδικασία της μέτρησης. Στο πείραμα EPR λοιπόν, έχουμε δύο σωματίδια (για παράδειγμα θα μπορούν να είναι ηλεκτρόνια) που βρίσκονται σε αλληλεπίδραση με αντίθετα spin που έχουν άθροισμα 0. Είναι εφικτό με κατάλληλες συνθήκες πειράματος να γνωρίζουμε το άθροισμα των ιδιοτήτων τους χωρίς τη γνώση των επιμέρους ιδιοτήτων, δηλαδή του spin στη προκειμένη περίπτωση. Με τον διαχωρισμό δύο ηλεκτρονίων σε μακρυνές αποστάσεις, αφού μετρηθεί η τιμή της ιδιότητας στο ένα ηλεκτρόνιο, τότε γνωρίζουμε τη τιμή στο απομακρυσμένο ακόμα και χωρίς μέτρηση, λόγω της γνώσης ότι το άθροισμά τους είναι σταθερό και μηδέν. Η παραδοξότητα έγκειται στο γεγονός ότι με τη μέτρηση (δηλαδή την «επιλογή») της τιμής του spin του ηλεκτρονίου, ταυτόχρονα «επιλέγουμε» και τη τιμή του spin του απομακρυσμένου ηλεκτρονίου. Με το γεγονός της κυματικής κατάρρευσης του πρώτου, υπάρχει κατάρρευση κύματος και στο δεύτερο[18,21].

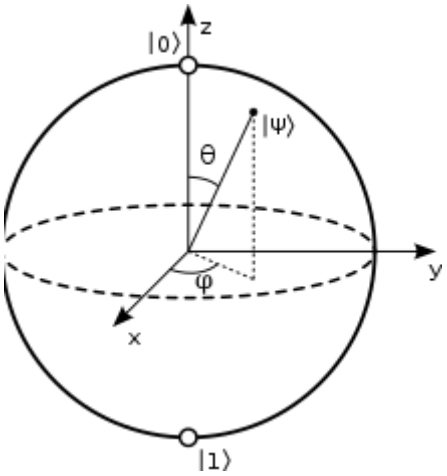
3.5 Κβαντική Υπολογιστική

Η ιδέα για τη δημιουργία ενός υπολογιστή με δομή βασισμένη στις αρχές της κβαντομηχανικής, διατυπώθηκε στις αρχές της δεκαετίας του ’90 από τους φυσικούς Richard Feynman, David Deutsch και Paul Benioff. Κύρια αφορμή αποτέλεσε η διαπίστωση ότι οι κλασικοί υπολογιστές παρουσιάζουν θεμελιώδεις περιορισμούς σχετικά με την πολυπλοκότητα χώρου και χρόνου προς επιτέλεση βασικών λειτουργιών. Κβαντικός υπολογιστής λοιπόν, ονομάζεται μία υπολογιστική μηχανή που εκμεταλλευόμενη βασικές ιδιότητες της κβαντικής μηχανικής (όπως την αρχή της επαλληλίας και της κβαντικής διεμπλοκής) φέρνει εις πέρας την επιτέλεση υπολογισμών και την επεξεργασία δεδομένων. Οι κλασικοί ψηφιακοί υπολογιστές αναπαριστούν τη πληροφορία οργανωμένη σε δυαδικά ψηφία (bits) με τιμές 0 και 1. Στους κβαντικούς υπολογιστές γίνεται το αντίστοιχο με τα quantum bits (qubits) με τις καταστάσεις να συμβολίζονται ως $|\mathbf{0}\rangle$ και $|\mathbf{1}\rangle$. Αξιοποιώντας τη κβαντική υπέρθεση, στους κβαντικούς υπολογιστές ένα qubit μπορεί να βρίσκεται ταυτόχρονα στις καταστάσεις $|\mathbf{0}\rangle$ και $|\mathbf{1}\rangle$ (σε αντίθεση με τους κλασικούς υπολογιστές). Ωστόσο, από τη στιγμή που θα πραγματοποιηθεί μία μέτρηση θα υπάρξει κατάρρευση στη κβαντική κατάσταση του qubit και θα επιστρέψει σε μία εκ των δύο βασικών καταστάσεων. Στη φυσική η περιγραφή ενός qubit μπορεί να πραγματοποιηθεί είτε ως το spin του ηλεκτρονίου είτε ως η πολικότητα του φωτονίου. Για παράδειγμα ως κατάσταση $|\mathbf{0}\rangle$ ορίζουμε την κατάσταση όπου το spin του ηλεκτρονίου είναι «κάτω», ενώ $|\mathbf{1}\rangle$

όταν είναι «πάνω». Μία ιδιαίτερα χρήσιμη αναπαράσταση για την ευκολότερη απεικόνιση των χαρακτηριστικών των qubits είναι η σφαίρα του Bloch. Αυτή παριστάνει ένα μεμονωμένο διάνυσμα κατάστασης $|\boldsymbol{\psi}\rangle$ το οποίο έχει μήκος ίσο με τη μονάδα και η αρχή του ξεκινάει από το κέντρο της σφαίρας και το βέλος εφάπτεται στην εσωτερική επιφάνειά της. Μία γενική μορφή αναπαράστασής της είναι η εξής:

$$|\boldsymbol{\psi}\rangle = \boldsymbol{\cos\frac{\theta}{2}}\,|\boldsymbol{0}\rangle + \boldsymbol{e^{i\varphi}}\ \boldsymbol{\sin\frac{\theta}{2}}\,|\boldsymbol{1}\rangle$$

Οι αριθμοί φ και θ είναι πραγματικοί και αναπαριστούν γωνίες, πιο συγκεκριμένα η θ ορίζει τις τιμές των πλατών πιθανότητας, ενώ η φ απεικονίζει τη γωνία φάσης:



Εικόνα 3.1: Σφαίρα Bloch [23]

Η γωνία φάσης διαφοροποιεί δύο διανύσματα τα οποία έχουν ίδια πλάτη πιθανότητα (αν και δεν μπορεί κανείς να το διακρίνει πρακτικά)[09,21,24].

Έστω λοιπόν ότι θέλουμε να αναπαραστήσουμε την κατάσταση ενός συστήματος με n qubits σε έναν κλασικό υπολογιστή. Θα χρειαστεί να αποθηκεύσουμε 2^n συντελεστές. Γίνεται άμεσα αντιληπτό οτι τα qubits έχουν τη δυνατότητα να αποθηκεύσουν εκθετικά περισσότερη πληροφορία συγκριτικά με τα bits. Ωστόσο, λαμβάνοντας υπ'όψη τη πιθανολογική υπέρθεση όλων των πιθανών καταστάσεων, όταν μετρήσουμε την τελική τους κατάσταση θα βρίσκονται μόνο σε έναν από τους πιθανούς σχηματισμούς που βρισκόνταν προ μετρήσεως. Ένα ενδεικτικό παράδειγμα ώστε να γίνει κατανοητό αυτό: δοθέντος συμβατικού υπολογιστή

που λειτουργεί σε καταχωρητή τριών bits, η κατάσταση του υπολογιστή είναι κάθε στιγμή πιθανότητα κατανεμημένη σε $2^3 = 8$ διαφορετικές ακολουθίες τριών bits (*000*, *001* κ.ο.κ.). Σε περίπτωση ντετερμινιστικού υπολογιστή, θα πρέπει να βρίσκεται σε μία από αυτές τις καταστάσεις με πιθανότητα *1*. Αν είναι πιθανολογικός μπορεί να βρίσκεται σε πληθώρα καταστάσεων που μπορεί να περιγραφούν με οχτώ μη αρνητικούς αριθμούς *A,B,C,D,E,F,G,H* (με την Α να αναπαριστά το 000, την Β το 001 κ.ο.κ.). Το άθροισμα των πιθανοτήτων είναι 1. Η κατάσταση του κβαντικού υπολογιστή των τριών bits περιγράφεται από διάνυσμα οχτώ διαστάσεων (a,b,c,d,e,f,g,h) που θα πρέπει το άθροισμα των τετραγώνων των συντελεστών να είναι ίσο με τη μονάδα ($|a|^2 + |b|^2 + |c|^2 + |d|^2 + |e|^2 + |f|^2 + |g|^2 + |h|^2 = 1$). Το απόλυτο τετράγωνο των συντελεστών υποδηλώνει το πλάτος πιθανότητας των δοθέντων καταστάσεων, ενώ αναπαριστά σημαντική παράμετρο η φάση μεταξύ δύο οποιονδήποτε συντελεστών (καταστάσεων)[14].

Κεφάλαιο 4
Κβαντική Κρυπτογραφία

4.1 Εισαγωγή στη κβαντική κρυπτογραφία

Όπως έχει ήδη αναλυθεί κανένα κλασσικό κρυπτοσύστημα δεν είναι απολύτως ασφαλές. Ανέκαθεν το κύριο ζήτημα κάθε συστήματος είναι η διαμοίραση του κρυφού κλειδιού μεταξύ των δύο χρηστών με τον ασφαλέστερο τρόπο. Με τη διαμοίραση, ωστόσο, συμπεριλαμβάνεται και η παραγωγή ενός πραγματικά τυχαίου μυστικού κλειδιού. Τα κβαντικά κρυπτογραφικά συστήματα αναπτύχθηκαν προς επίλυση αυτού του προβλήματος, κατά τα οποία αφού πραγματοποιηθεί η διαδικασία δημιουργίας και διαμοίρασης του μυστικού κλειδιού, το κβαντικό μέρος έχει λάβει τέλος, και έπεται η κλασική διαδικασία κρυπτογράφησης, αποστολής και αποκρυπτογράφησης των δεδομένων[09]. Δηλαδή η κβαντική κρυπτογραφία εγγυάται πλήρη μυστικότητα και ασφάλεια καθώς στηρίζεται στους νόμους της φυσικής, σε αντίθεση με τη κλασική κρυπτογραφία που θα μπορούσε να παραβιαστεί από έναν πολύ γρήγορο υπολογιστή. Παρά το γεγονός ότι η ιδέα προτάθηκε πρώτη φορά τη δεκαετία του 1970, δεν εφαρμόστηκε στην αφάλεια πληροφοριών μέχρι τις αρχές της δεκαετίας του 1990 [22]. Για την επιτυχία της στην

επίλυση του προβλήματος διανομής κλειδιού, χρησιμοποιούνται τα φωτόνια μέσω οπτικών ινών ή ελεύθερου χώρου, καθώς αυτά τηρούν την αρχή της αβεβαιότητας του Heisenberg ή την κβαντική διεμπλοκή. Η πρώτη σχετικά με τις ιδιότητες των φωτονίων με βάση τις οποίες όταν κωδικοποιούνται ειδικές πληροφορίες στις ιδιότητες ενός φωτονίου, κάθε απόπειρα παρακολούθησης ενός φωτονίου αλλάζει τις ιδιότητές του ώστε η απόπειρα παρακολούθησης να γίνεται αντιληπτή από τον αποδέκτη του μηνύματος (με τη μέτρηση του φωτονίου, αυτόματη αλλαγή κατάστασης)[24]. Ένας τρόπος μετάδοσης κβαντικού κλειδιού κρυπτογράφησης απαιτεί την ύπαρξη λέιζερ που θα μπορεί να εκπέμπει μεμονωμένα φωτόνια που θα είναι πολωμένα με δύο διαφορετικούς τρόπους. Κατά τον πρώτο τρόπο τα φωτόνια θα έχουν κατακόρυφη ή οριζόντια πόλωση (ορθός τρόπος), ενώ με τον δεύτερο θα σχηματίζει με την κατακόρυφο γωνία $\pm 45°$ (πλάγιος τρόπος). Ένα ζεύγος καταστάσεων ορθογώνιας (κάθετης) πόλωσης που χρησιμοποιούνται για να περιγράψουν την πόλωση φωτονίων, όπως οριζόντια / κάθετη, αναφέρεται ως βάση. Και στις δύο περιπτώσεις οι δύο αμοιβαίως ορθογώνιες πολώσεις αναπαριστούν το 0 η μία και το ψηφίο 1 η άλλη. Ο αποστολέας με έναν τυχαίο τρόπο απο τους δύο προηγούμενους (ορθό και πλάγιο) για μεταφορά φωτονίων, και θα αποστείλλει έτσι μία ακολουθία ψηφίων (bits). Αντίστοιχα ο αποδέκτης θα επιλέξει τυχαία τον τρόπο τον οποίο θα χρησιμοποιήσει ώστε να μετρήσει να εισερχόμενα bits. Θα είναι ικανός όμως να τα μετρήσει μόνο με έναν από τους δύο τρόπους λόγω της αρχής της αβεβαιότητας του Heisenberg. Μόνο τα φωτόνια που μέτρησε ο παραλήπτης με τον ίδιο τρόπο που μεταδόθηκαν από τον αποστολέα είναι βέβαιο πως θα έχουν την ίδια πόλωση και θα συμπίπτουν τα bits. Μετά την αποστολή θα χρειαστεί ο παραλήπτης να επικοινωνήσει με τον αποστολέα (κρυφά) ώστε να τον ενημερώσει ποιον από τους δύο τρόπους χρησιμοποίησε για να λάβει το κάθε φωτόνιο, χωρίς να αναφέρει ποια από τις δύο δυαδικές τιμές αναπαριστά το κάθε φωτόνιο. Έπειτα ο αποστολέας θα αποκαλύψει στον παραλήπτη ποια φωτόνια μετρήθηκαν σωστά, ενώ όσα μετρήθηκαν λάθος αγνοούνται και από τους δύο. Το κλειδί που θα χρησιμοποιηθεί από τον αλγόριθμο για κρυπτογράφηση και αποκρυπτογράφηση του μηνύματος θα είναι τα σωστά μετρημένα bits. Η αρχή της απροσδιοριστίας απαγορεύει σε έναν πιθανό τρίτο υποκλοπέα να χρησιμοποιήσει και τους δύο τρόπους μετρήσεων ώστε να κλέψει το κλειδί. Έτσι, παραλήπτης και αποστολέας μέσω σύγκρισης των bits μπορούν να ανακαλύψουν σφάλματα τα οποία μεταφράζονται σε προσπάθεια υποκλοπής[24].

4.2 Πρωτόκολλο BB84

Το BB84 πρωτόκολλο αποτέλεσε μια πρώτη προσέγγιση για τη δημιουργία ενός κβαντικού κρυπτογραφικού πρωτόκολλου διαμοίρασης μυστικού κλειδιού, από τους Charles Bennet και Gilles Brassard, το 1984 .Με βάση τα προηγούμενα θα περιγραφεί η διαδικασία του χρησιμοποιώντας το κλασικό παράδειγμα με την Alice και τον Bob να επιθυμούν να επικοινωνήσουν και την Eve να προσπαθεί να υποκλέψει πληρφορίες από την επικοινωνία τους. Υπάρχει το στάδιο της κβαντικής επικοινωνίας:

a) Η Alice στέλνει στον Bob μια σειρά από φωτόνια κάθε ένα επιλεγμένο ανεξάρτητα με μία από τις τέσσερις πολώσεις (οριζόντια, κατακόρυφη, 45 μοιρών, 135 μοιρών).

b) Για κάθε φωτόνιο ο Bob επιλέγει μία από τις δύο βάσεις (ορθή και πλάγια) για να πραγματοποιήσει τη μέτρηση.

c) Ο Bob καταγράφει τις μετρήσεις των βάσεων και τα αποτελέσματα. Αναφέρει δημόσια τα σήματά του.

Καθώς και το στάδιο της δημόσιας επικοινωνίας:

A. Η Alice μεταδίδει τις βάσεις των μετρήσεών της. Ο Bob μεταδίδει τις βάσεις των μετρήσεών του.

B. Η Alice και ο Bob απορρίπτουν όλα τα συμβάντα όπου χρησιμοποιούν διαφορετικές βάσεις για ένα σήμα.

C. Για να ελέγξει την παραβίαση, η Alice επιλέγει τυχαία ένα μέρος από όλα τα υπόλοιπα συμβάντα ως δοκιμαστικά γεγονότα. Για εκείνα τα δοκιμαστικά γεγονότα, μεταδίδει δημόσια τις θέσεις και την πόλωσή τους.

D. Ο Bob μεταδίδει τις πολώσεις όλων των δοκιμαστικών γεγονότων.

E. Η Alice και ο Bob υπολογίζουν το ποσοστό σφάλματος των γεγονότων δοκιμής. Εάν το υπολογισμένο ποσοστό σφάλματος είναι μεγαλύτερο από κάποιο προβλεπόμενο με τιμή κατωφλίου, ας πούμε 11%, ματαιώνουν. Διαφορετικά, προχωρούν στο επόμενο βήμα.

F. Η Alice και ο Bob μετατρέπουν ο καθένας τα δεδομένα πόλωσης όλων των δεδομένων που απομένουν σε μία δυαδική συμβολοσειρά που ονομάζεται ακατέργαστο κλειδί (για παράδειγμα, αντιστοιχίζοντας ένα κατακόρυφο φωτόνιο ή ένα φωτόνιο 45 μοιρών σε "0" και ένα οριζόντιο φωτόνιο ή 135 μοιρών στο "1"). Μπορούν να εκτελέσουν κλασική μετα-επεξεργασία, όπως διόρθωση σφαλμάτων και ενίσχυση απορρήτου για να δημιουργήσουν ένα τελικό κλειδί.

Σημαντική επισήμανση αποτελεί το γεγονός ότι είναι σημαντικό το κλασσικό κανάλι επικοινωνίας μεταξύ της Alice και του Bob να είναι πιστοποιημένο. Διαφορετικά, η Eve μπορεί εύκολα να εξαπολύσει μια επίθεση man-in-the-middle μεταμφιεσμένη σε Alice στον Bob και ως Bob στην Alice. Ευτυχώς, η ταυτοποίηση ενός κλασσικού μηνύματος m-bit απαιτεί μόνο λογαριθμική σε m bit ενός κλειδιού ελέγχου ταυτότητας. Επομένως, η κβαντική διαμοίραση κλειδιού παρέχει έναν αποτελεσματικό τρόπο επέκτασης ενός σύντομου αρχικού κλειδιού ελέγχου ταυτότητας σε ένα μακρύ κλειδί. Επαναλαμβάνοντας τη διαδικασία πολλές φορές, μπορεί κανείς να λάβει ένα αυθαίρετα μεγάλο ασφαλές κλειδί[25].

Εφόσον η Eve αδυνατεί να κάνει αντίγραφα της ακολουθίας και δεδομένου ότι μετά τη παρεμπόδιση πρέπει να στείλει κάτι στον Bob, το καλύτερο που μπορεί να πράξει είναι να μετρά κάθε bit που λαμβάνει και να στέλνει έπειτα ένα διαφορετικό φωτόνιο που προετοιμάζεται στην ίδια κατάσταση με τα αποτελέσματα μέτρησής της. Με αυτόν τον τρόπο θα είχε 50% πιθανότητα να στείλει την ίδια κατάσταση με αυτήν που έστειλε η Alice. Σε τέτοιες περιπτώσεις ο Bob και η Alice δε θα είναι σε θέση να ανακαλύψουν την επέμβασή της. Άρα στο υπόλοιπο 50% των περιπτώσεων θα διαλέξει αντίθετη βάση που θα εισαγάγει πρόσθετο λάθος στην ακολουθία που λαμβάνεται από τον Bob. Αυτό θα οδηγήσει σε πρόσθετο λάθος 25% στην ακολουθία του Bob, και αφού αυτός συγκρίνει τις βάσεις με την Alice θα ανιχευόταν η επέμβαση της Eve[26]. Επίσης, όταν κανένας υποκλοπέας δεν προκαλεί αλλαγή bit, θα υπάρχουν στην πράξη κάποια σφάλματα στη μετάδοση και η συμβολοσειρά της Alice και του Bob δεν θα συμπίπτουν τέλεια. Τα υπόλοιπα σφάλματα αφαιρούνται με τυπικές μεθόδους διόρθωσης σφαλμάτων, η οποία με τη σειρά της μειώνει το μήκος του κλειδιού. Το καθορισμένο ποσοστό σφάλματος (συνήθως της τάξης του 1%) πρέπει να αποδοθεί πλήρως στην Eve. Οι αντίστοιχες πληροφορίες που μπορεί να αποκτήσει η Eve μπορούν να μειωθούν σε αυθαίρετα χαμηλά

τιμή, με μια διαδικασία που ονομάζεται ενίσχυση απορρήτου, και πάλι στο κόστος του μήκους του χρησιμοποιήσιμου κλειδιού. Επομένως, είναι πολύ σημαντικό να διατηρηθεί το ποσοστό πειραματικού σφάλματος όσο δυνατόν χαμηλά [27]. Εάν το ποσοστό σφάλματος είναι κάτω από μια συμφωνημένη τιμή κατωφλίου, η Alice και ο Bob μπορούν να εξαλείψουν τα σφάλματα με (κλασσική) διόρθωση σφαλμάτων. Μια απλή μέθοδος διόρθωσης σφαλμάτων λειτουργεί ως εξής: Η Alice επιλέγει δύο bit τυχαία και λέει στον Bob την τιμή XOR των δύο bit. Ο Bob λέει στην Alice αν έχει την ίδια αξία. Σε αυτή την περίπτωση, κρατούν το πρώτο bit και απορρίπτουν το δεύτερο bit. Εάν οι τιμές τους διαφέρουν, τότε απορρίπτονται και τα δύο bit. Τα υπόλοιπα bits αποτελούν το κλειδί[28]. Θα χρησιμοποιήσουμε το παρακάτω παράδειγμα για να γίνει το πρωτόκολλο πιο κατανοητό. Παρακάτω φαίνονται οι καταστάσεις που χρησιμοποιούμε για την αναπαράσταση του δυαδικού ψηφίου 0 καθώς και του δυαδικού ψηφίου 1:

Basis	0	1
+	↑	→
×	↗	↘

Εικόνα 4.1: Πίνακας αντιστοίχισης βάσεων σε δυαδικά ψηφία[29]

Όπως φαίνεται και στο παρακάτω σχήμα η Alice και ο Bob συμφωνούν στις μισές βάσεις:

Alice's random bit	0	1	1	0	1	0	0	1
Alice's random sending basis	+	+	×	+	×	×	×	+
Photon polarization Alice sends	↑	→	↘	↑	↘	↗	↗	→
Bob's random measuring basis	+	×	×	×	+	×	+	+
Photon polarization Bob measures	↑	↗	↘	↗	→	↗	→	→
PUBLIC DISCUSSION OF BASIS								
Shared secret key	0		1			0		1

Εικόνα 4.2: **Παράδειγμα πρωτοκόλλου BB84 προς δημιουργία κοινού μυστικού κλειδιού**[29]

Η Alice θα διαλέξει μια κατάσταση πόλωσης να στείλει στον Bob ο οποίος επίσης θα επιλέξει μία βάση. Αφού γίνει υπολογισμός των βάσεων μέσω ενός δημόσιου καναλιού, απορρίπτουν όλα τα bits τα οποία υπολογίστηκαν με διαφορετικες βάσεις. Κάποια από τα υπόλοιπα bits μπορούν να χρησιμοπιηθούν ώστε να γίνει έλεγχος της Eve και επίσης να απορριφθούν. Τα εναπομείναντα bits θα αποτελέσουν το τελικό μυστικό κλειδί[22,29].

4.3 Υλοποίηση BB84

Για την μοντελοποίηση και εφαρμογή του πρωτοκόλλου BB84 έγινε χρήση της γλώσσας προγραμματισμού Python και υλοποίηση στο περιβάλλον του Google Colab. Χρησιμοποιήθηκε ο κώδικας από το [30] αποθετήριο github. Αρχικά εισάγουμε το module random που υλοποιεί γεννήτριες ψευδοτυχαίων για διάφορες κατανομές. Έπειτα κατασκευάζουμε μία κλάση με το όνομα Alice. Αυτή θα περιέχει τα διάφορα bits, τη βάση της (2 περιπτώσεις) και τις πολώσεις (4 περιπτώσεις). Στις δύο περιπτώσεις για τη βάση (ορθή και πλάγια) γίνεται αντιστοίχιση επίσης σε δυαδικά ψηφία με το 0 να αντικατοπτρίζει την ορθή βάση και τη πλάγια το 1. Έπειτα γίνεται κατασκευή μιας συνάρτησης η οποία να μπορεί να κάνει reset τα προηγούμενα. Επιπρόσθετα δημιουργείται συνάρτηση για τη δημιουργία τυχαίων bits και βάσης από γεννήτριες ψευδοτυχαίων αριθμών. Στη συνέχεια κατασκευάζεται συνάρτηση και για την αντίστοιχη περίπτωση συνδυασμού bit και βάσης θέτει την κατάλληλη πόλωση (-45,0,45,90 μοίρες):

```
import random

class Alice:
    bits=[] #0,1
    basis=[] #+(0),x(1)
    polarization=[] #0,90,45,-45

    def reset(self):#δημιούργησε μία reset συνάρτηση για να είναι ευκολότερο να μετράει πολλές φορές
        self.bits=[]
        self.basis=[]
        self.polarization=[]

    def create_bits(self, num_bits): #το num_bits αποτελεί είσοδο από την main
        for i in range(0,num_bits,1):
            self.bits.append(random.randint(0,1))#τα bits δημιουργούνται τυχαία και είναι τόσα όσο το πληθος των bits που στέλνονται
            self.basis.append(random.randint(0,1))#η βάση επίσης επιλέγεται τυχαία
        #print("alice παρήγαγε : ",self.bits)
        #print("alice παρήγαγε basis : ",self.basis)

    def measure_polarization(self):#μέτρηση πόλωσης
        for i in range(0,len(self.bits),1):
            if self.bits[i]==0 and self.basis[i]==0 :
                self.polarization.append(90)
            elif self.bits[i]==0 and self.basis[i]==1:
                self.polarization.append(45)
            elif self.bits[i]==1 and self.basis[i]==0:
                self.polarization.append(0)
            elif self.bits[i]==1 and self.basis[i]==1:
                self.polarization.append(-45)
        #print("alice δημιούργησε πόλωση : ",self.polarization)
        return self.polarization
```

Έπειτα γίνεται κατασκευή δύο συναρτήσεων για έλεγχο αν τα bits και η βάση είναι ίδια με του Bob:

```
def send_bits(self):#χρειάζεται έλεγχος αν ειναι ίδια τα bits με του Bob
    return self.bits

def send_basis(self):#χρειάζεται έλεγχος αν ειναι ίδια η βάση με του Bob
    return self.basis
```

Του οποίου τη κλάση θα παράξουμε στη συνέχεια. Θα δημιουργηθούν οι ίδιες λίστες (bits, βάσεων, πολώσεων) με της Alice με μία επιπρόσθετη τα bits που θα ανταλλαχτούν (που προφανώς θα προστεθεί και στη reset συνάρτηση). Η επόμενη συνάρτηση θα έχει ως στόχο τη τυχαία

παραγωγή βάσης από τον Bob για τα bits που θα δέχεται από την Alice:

```
class Bob:
    bits=[] #0,1
    basis=[] #+(0),x(1)
    real_bits=[] #bits που χρησιμοποιούνε την ίδια βάση
    exchange=[]

    def reset(self):#δημιούργησε μία reset συνάρτηση για να είναι ευκολότερο να μετράει πολλές φορές
        self.bits=[]
        self.basis=[]
        self.real_bits=[]
        self.exchange=[]

    def create_basis(self,num_bits):#num_bits ίδια με της  Alice
        for i in range(0,num_bits,1):
            self.basis.append(random.randint(0,1))#τυχαία επιλογή από τη βάση
        #print("βάση που παράγεται από τον Bob : ",self.basis)
```

Στη συνέχεια, αφού λάβει τη πόλωση από την Alice ως παράμετρο για κάθε περίπτωση των 4 διαφορετικών πολώσεων που μπορεί να λάβει σε συνδυασμό με τις δύο περιπτώσεις βάσεων που μπορεί να έχει, θα αποθηκεύει το κατάλληλο bit που πρέπει να αντιστοιχίζεται:

```
def measure_bits(self,received):#Λαμβάνει τη πόλωση που στέλνεται από την Alice/Eve ως παράμετρος
    for i in range(0,len(self.basis),1):
        if self.basis[i]==0 and received[i]==90 :
            self.bits.append(0)
        elif self.basis[i]==0 and received[i]==0:
            self.bits.append(1)
        elif self.basis[i]==1 and received[i]==45:
            self.bits.append(0)
        elif self.basis[i]==1 and received[i]==-45:
            self.bits.append(1)
        elif self.basis[i]==0 and received[i]==45:
            self.bits.append(random.randint(0,1))
        elif self.basis[i]==0 and received[i]==-45:
            self.bits.append(random.randint(0,1))
        elif self.basis[i]==1 and received[i]==90:
            self.bits.append(random.randint(0,1))
        elif self.basis[i]==1 and received[i]==0:
            self.bits.append(random.randint(0,1))
    #print("bits που μετρούνται από τον Bob : ",self.bits)
```

Τέλος, γίνεται μέτρηση των ανταλλάξιμων (bits των οποίων «συμφώνησα») bits σε σχέση με όλα τα bits που στάλθηκαν και γίνεται υπολογισμός του ρυθμού σφάλματος:

```
def measure_real_bits(self,received,bits):#Λαμβάνει τη βάση που στέλνεται από την Alice/Eve ως παράμετρος
    for i in range(0,len(self.basis),1):
        if received[i]==self.basis[i]:
            if bits[i]==self.bits[i]:
                self.exchange.append(self.bits[i])
            self.real_bits.append(self.bits[i])
    #print("ανταλλάξιμα bits : ",self.real_bits)
    #print("πραγματικά ανταλλάξιμα bits : ",self.exchange)
    #print("πραγματικά ανταλλάξιμα bits/αυθεντικός αριθμός bits",len(self.exchange)/len(bits))
    #return len(self.exchange)/len(bits)
    print("QBER(Quantum Bit Error Rate) : ",(len(self.real_bits)-len(self.exchange))/len(self.real_bits)*100,"%")
    return (len(self.real_bits)-len(self.exchange))/len(self.real_bits)*100
```

Και η τελευταία κλάση που μένει να δημιουργηθεί είναι αυτή της Eve που θα έχει λίστες για βάση και πόλωση με επιπρόσθετη μία λίστα για τη πόλωση της Alice:

```
class Eve:
    basis=[] #+(0),x(1)
    polarization=[] #0,90,45,-45
    a_polarization=[]

    def reset(self):#δημιούργησε μία reset συνάρτηση για να είναι ευκολότερο να μετράει πολλές φορές
        self.basis=[]
        self.polarization=[]
        self.a_polarization=[]

    def create_basis(self,num_bits): #το num_bits αποτελεί είσοδο από την main
        for i in range(0,num_bits):
            self.basis.append(random.randint(0,1))#τυχαία επιλογή από τη βάση
        #print("βάση παραγόμενη από την Eve : ",self.basis)
```

Έπειτα φτάνει το μέρος που ξεκινάει να λετουργεί ως λαθρακροάτρια δεχόμενη τη βάση που στέλνεται από την Alice ως παράμετρο θα προσπαθεί να υποκλέψει τα bits που στέλνει, τα οποία ωστόσο θα αλλάζει σε κάθε λανθασμένη περίπτωση: :

```
def measure_bits(self,received,eavesdropping):#Λαμβάνει τη βάση που στέλνεται από την Alice ως παράμετρο
    for i in range(0,int(len(self.basis)*eavesdropping),1):#το σημείο όπου υπάρχει λαθρακουστής
        if self.basis[i]==0 and received[i]==90 :
            self.polarization.append(90)
        elif self.basis[i]==0 and received[i]==0:
            self.polarization.append(0)
        elif self.basis[i]==1 and received[i]==45:
            self.polarization.append(45)
        elif self.basis[i]==1 and received[i]==-45:
            self.polarization.append(-45)
        elif self.basis[i]==0 and received[i]==45:
            if random.randint(0,1)==0:
                self.polarization.append(0)
            else :
                self.polarization.append(90)
        elif self.basis[i]==0 and received[i]==-45:
            if random.randint(0,1)==0:
                self.polarization.append(0)
            else :
                self.polarization.append(90)
        elif self.basis[i]==1 and received[i]==90:
            if random.randint(0,1)==0:
                self.polarization.append(45)
            else :
                self.polarization.append(-45)
        elif self.basis[i]==1 and received[i]==0:
            if random.randint(0,1)==0:
                self.polarization.append(45)
            else :
                self.polarization.append(-45)
    #print("πόλωση μετρημένη από την Eve: ",self.polarization)
```

Τέλος λαμβάνουμε υπ'όψη και το μέρος χωρίς λαθρακρόαση και κατασκευάζουμε συναρτήσει για αποστολή στο κανάλι και λήψη πόλωσης από την Alice:

```
    for i in range(int(len(self.basis)*eavesdropping),len(self.basis),1):#μέρος χωρίς λαθρακουστή
        self.polarization.append(self.a_polarization[i])

def send_polarization(self): #αποστολή στο κανάλι
    return self.polarization

def receive_p(self,alice):#Λήψη πόλωσης της Alice
    self.a_polarization=alice
```

Πλέον μπορεί να ξεκινήσει η διαδικασία για το παράδειγμα εφαρμογής του πρωτοκόλλου BB84 με την αντίστοιχη μέτρηση σφαλμάτων. Θα χρειαστεί να εισάγουμε τη βιβλιοθήκη matplotlib για

σχεδίαση και οπτικοποίηση των αποτελεσμάτων και την pandas για τη διαχείριση δεδομένων. Με την χρήση των κλάσεων που κατασκευάσαμε νωρίτερα φτιάχνουμε ένα αντίστοιχο παράδειγμα για κάθε κλάση (Alice,Bob και Eve). Κατόπιν, αρχικοποιούμε μεταβλητές που θα χρησιμοποιηθούν στη συνέχεια. Δημιουργούμε μία κένη λίστα για τη πόλωση που στέλνεται από το κβαντικό κανάλι, και άλλη μία μεταβλητή άξια επισήμανσης είναι ο ρυθμός με τον οποίο συμβαίνει η λαθρακρόαση που αρχικοποιείται με 0.1 (10%). Οι υπόλοιπες μεταβλητές σχετίζονται με τα bits, τις βάσεις και τον ρυθμό σφάλματος:

```
import random
from matplotlib import pyplot as plt
import pandas as pd

alice=Alice()
bob=Bob()
eve=Eve()
num_bits=[500]
bits=[]
basis=[]
qc_polarization=[]#πόλωση που στέλνεται από το κβαντικό κανάλι
count=0
QBER=[]
sum_QBER=0
sum_data=0
eavesdropping_rate=0.1
```

Έπειτα γίνεται η υλοποίηση της κύριας διαδικασίας κατά την οποία ώστε να ξεκινήσει παράγουμε τυχαίους αριθμούς και κάνουμε reset και στα τρία αντικείμενα:

```
#print("αριθμός bits: ",num_bits[i])

while(eavesdropping_rate<=1):
    random.seed()

    alice.reset()
    bob.reset()
    eve.reset()

    #int(input("πόσα bits να δημιουργηθούν? "))
```

Σε αυτήν με χρήση των προηγούμενων συναρτήσεων που έχουν δημιουργηθεί και την κατάλληλη αξιοποίησή τους φαίνεται η παραγωγή και αποστολή bits της Alice, η παρέμβαση της Eve, η λήψη των bits από τον Bob και τέλος ο έλεγχος αντιστοιχίας των bits μεταξύ Alice και Bob με σημείο αναφοράς τις βάσεις που χρησιμοποιούσαν. Η διαδικασία επαναλαμβάνεται 10 φορές με ρυθμό αύξησης του ρυθμού λαθρακρόασης 10% για κάθε κύκλο επανάληψης:

```
alice.create_bits(num_bits[i]) #Η Alice παράγει bits και τυχαία βάση
qc_polarization=alice.measure_polarization() #πόλωση μετρημένη από την Alice σε bit και βάση
eve.receive_p(qc_polarization)#μέρος όπου δεν υπάρχει λαθρακουστής με την πόλωση της Alice

eve.create_basis(num_bits[i]) # Η Eve επιλέγει τη βάση τυχαία
eve.measure_bits(qc_polarization,eavesdropping_rate) #μέτρηση πόλωσης στη βάση της Eve
qc_polarization=eve.send_polarization() #Μετάδοση του αλλαγμένου σήματος στο κβαντικό κανάλι

bob.create_basis(num_bits[i]) #Ο Bob τυχαία επιλέγει τη βάση
bob.measure_bits(qc_polarization) #μέτρηση bits με βάση και πόλωση

basis=alice.send_basis() #Η Alice στέλνει βάση στο κλασικό κανάλι
bits=alice.send_bits() #Η Alice στέλνει bits

QBER.append(bob.measure_real_bits(basis,bits)) #Ελέγχει αν είναι ίδια η βάση με της Alice και αποθηκεύει το πραγματικό bit

eavesdropping_rate+=0.1
```

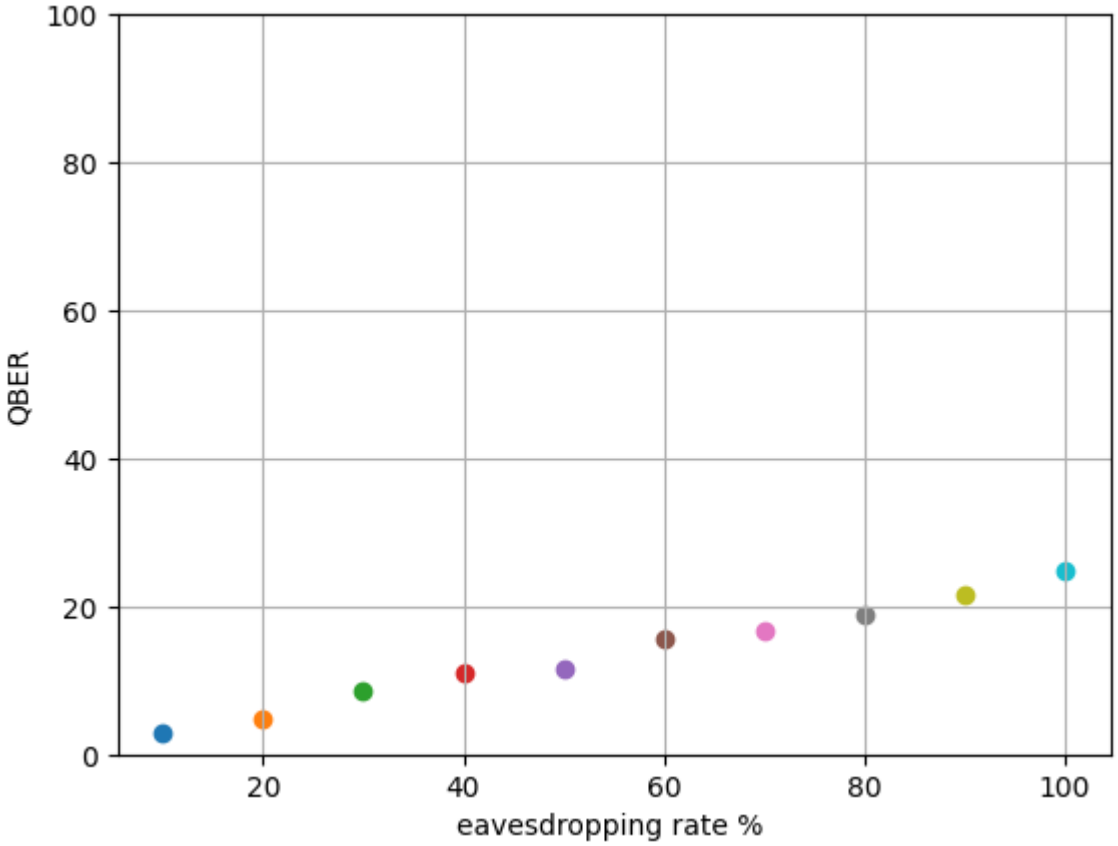
Τέλος, πραγματοποιείται η κατασκευή δύο αξόνων, ενός κάθετου που αντικατοπτρίζει το QBER (quantum bit error rate-ρυθμός σφάλματος κβαντικών bits) και ενός οριζόντιου που αντιστοιχεί στον ρυθμό λαθρακρόασης, που αντιπροσωπεύει τον μέγιστο ρυθμό με τον οποίο ένας λαθρακροατής μπορεί να αποκτήσει πληροφορίες για το κβαντικό κλειδί χωρίς να ανιχνευθεί. Θα χρησιμοποιηθεί το QBER στην οπτικοποίηση του ρυθμού σφάλματος για κάθε αντίστοιχο ρυθμό λαθρακρόασης:

```
z0=pd.DataFrame({'x0' : 10,'y0' : QBER[0]},index=[0])
z1=pd.DataFrame({'x1' : 20,'y1' : QBER[1]},index=[1])
z2=pd.DataFrame({'x2' : 30,'y2' : QBER[2]},index=[2])
z3=pd.DataFrame({'x3' : 40,'y3' : QBER[3]},index=[3])
z4=pd.DataFrame({'x4' : 50,'y4' : QBER[4]},index=[4])
z5=pd.DataFrame({'x5' : 60,'y5' : QBER[5]},index=[5])
z6=pd.DataFrame({'x6' : 70,'y6' : QBER[6]},index=[6])
z7=pd.DataFrame({'x7' : 80,'y7' : QBER[7]},index=[7])
z8=pd.DataFrame({'x8' : 90,'y8' : QBER[8]},index=[8])
z9=pd.DataFrame({'x9' : 100,'y9' : QBER[9]},index=[9])

plt.scatter(z0['x0'],z0['y0'])
plt.scatter(z1['x1'],z1['y1'])
plt.scatter(z2['x2'],z2['y2'])
plt.scatter(z3['x3'],z3['y3'])
plt.scatter(z4['x4'],z4['y4'])
plt.scatter(z5['x5'],z5['y5'])
plt.scatter(z6['x6'],z6['y6'])
plt.scatter(z7['x7'],z7['y7'])
plt.scatter(z8['x8'],z8['y8'])
plt.scatter(z9['x9'],z9['y9'])

plt.xlabel('eavesdropping rate % ')
plt.ylabel('QBER')
plt.ylim([0,100])
plt.grid()
plt.show()
```

Τυπώνουμε τα αποτελέσματα του QBER για τον αντίστοιχο ρυθμό λαθρακρόασης με αύξηση του δευτέρου ανά 10%:

```
QBER(Quantum Bit Error Rate) :  2.8112449799196786 %
QBER(Quantum Bit Error Rate) :  4.8582995951417 %
QBER(Quantum Bit Error Rate) :  8.641975308641975 %
QBER(Quantum Bit Error Rate) :  11.11111111111111 %
QBER(Quantum Bit Error Rate) :  11.538461538461538 %
QBER(Quantum Bit Error Rate) :  15.53784860557769 %
QBER(Quantum Bit Error Rate) :  16.727272727272727 %
QBER(Quantum Bit Error Rate) :  18.803418803418804 %
QBER(Quantum Bit Error Rate) :  21.666666666666668 %
QBER(Quantum Bit Error Rate) :  24.896265560165975 %
```

Και παρατηρούμε ότι λαμβάνει μέγιστη τιμή περίπου 25% το QBER για μέγιστο ρυθμό λαθρακρόασης. Έπειτα οπτικοποιούμε τα QBER ως προς τον αντίστοιχο ρυθμό λαθρακρόασης με τη βοήθεια τον αξόνων που κατασκευάσαμε[30]:



4.4 Πρωτόκολλο B92

Εκτός από το πρωτόκολλο BB84 των τεσσάρων καταστάσεων που πρωτοεμφανίστηκε το '84, ο Charles Bennett βασιζόμενος στο αξίωμα ότι είναι αδύνατον να διαχωρίσεις δύο μη ορθογώνιες καταστάσεις, χωρίς να προκληθεί διατάραξη των καταστάσεων (βασίζεται επίσης στην αρχή της απροσδιοριστίας όπως το BB84), παρουσίασε την ιδέα ενός πρωτοκόλλου των 2 καταστάσεων κατά το '92. Δηλαδή σε αυτό χρησιμοποίησε δύο οποιεσδήποτε μη ορθογώνιες καταστάσεις (αντί για τέσσερις) το οποίο είναι και η βασική του διαφορά επί της ουσίας με το BB84, για αυτό και θεωρείται απλώς ως μία πιο απλοποιημένη εκδοχή του[10,24]. Η κωδικοποίηση πληροφοριών μεταξύ δύο μη ορθογώνιων καταστάσεων καθιστούν αδύνατο για τον λαθρακουστή να διακρίνει μεταξύ δύο κβαντικών καταστάσεων του συστήματος. Αποτελεί ένα πρωτόκολλο διανομής κβαντικού κλειδιού που επίσης χρησιμοποιεί φωτόνια ως διανομείς πληροφοριών και οι δύο νόμιμοι χρήστες (Alice και Bob) επικοινωνούν επίσης μέσω δύο καναλιών (που ο εχθρός έχει επίσης πρόσβαση):

- Ένα κλασσικό που μπορεί να είναι δημόσιο: Η Eve μπορεί να ακούσει παθητικά (χωρίς να ανιχνευθεί)

- Ένα κβαντικό όπου-εγγενώς- η Eve δε μπορεί να ακούσει παθητικά[32,33]

Η πρώτη φάση του B92 περιλαμβάνει μεταδόσεις μέσω κβαντικού καναλιού, ενώ η δεύτερη λαμβάνει χώρα μέσω του κλασικού. Τα βήματα του πρωτοκόλλου έχουν ως:

1) Η Alice παράγει μία τυχαία σειρά από bits

2) Η Alice επιλέγει τυχαία μία από τις δύο καταστάσεις πόλωσης (0 και 45 μοίρες). Έπειτα δημιουργεί μία κατάσταση πόλωσης φωτονίου η οποία αντιστοιχεί στη τιμή του bit που έχει επιλέξει.

3) Η Alice στέλνει τα qubits στον Bob μέσω του κβαντικού καναλιού.

4) Ο Bob λαμβάνει τα qubits και επιλέγει βάση για να μετρήσει το κάθε ένα από αυτά τυχαία (ορθώς ή πλαγίως)

5) Αν ο Bob ανιχνεύσει ένα φωτόνιο, γνωρίζει τη πόλωση και την τιμή του bit που μεταδόθηκε από την Alice, αλλά αν δε το κάνει δε διαθέτει κάποια ένδειξη για τη κατάσταση που αυτή έστειλε. Ως αποτέλεσμα ο Bob θα κρατήσει μόνο τις τιμές όπου ανιχνεύθηκε το φωτόνιο.

6) Ο Bob επικοινωνεί με την Alice μέσω ενός κλασικού καναλιού για τα ποια qubits μπορούσε να ανιχνεύσει το φωτόνιο, και αυτές οι αντίστοιχες τιμές bit θα αποτελέσουν το κλειδί.

7) Η Alice και ο Bob ανταλλάσουν ένα τυχαίο δείγμα των bit και ελέγχουν τα σφάλματα[34]

4.5 Υλοποίηση B92

Η υλοποίηση θα γίνει με τις ίδιες συνθήκες που έγινε και αυτή του BB84. Αρχικά θα δημιουργηθεί μία κλάση για την Alice χωρίς τη δημιουργία κενής λίστας για τη βάση, μίας συνάρτησης για reset και μιας για δημιουργία bits (με δύο περιπτώσεις αυτή τη φορά, για 0 και 45 μοίρες). Στο τέλος δυο συναρτήσεις για έλεγχο ομοιότητας bits με τον Bob καθώς και πόλωσης .Ομοίως χρησιμοποιήθηκε το αποθετήριο github που αναφέρεται στο [30]:.

```
import random

class Alice:
    bits=[] #0,1
    polarization=[] #0,90,45,-45

    def reset(self):#δημιούργησε μία reset συνάρτηση για να είναι ευκολότερο να μετράει πολλές φορές
        self.bits=[]
        self.polarization=[]

    def create_bits(self, num_bits): #το num_bits αποτελεί είσοδο από την main
        for i in range(0,num_bits,1):
            x=random.randint(0,1)
            self.bits.append(x)#τα bits δημιουργούνται τυχαία και είναι τόσα όσο το πληθος των bits που στέλνονται
            if x==0 :
                self.polarization.append(0)
            elif x==1:
                self.polarization.append(45)
        #print("alice παρήγαγε bits : ",self.bits)
        #print("πόλωση της Alice : ",self.polarization)
    def send_bits(self):#χρειάζεται έλεγχος αν ειναι ίδια τα bits με του Bob
        return self.bits

    def send_polarization(self):
        return self.polarization
```

Έπειτα δημιουργία κλάσης για τον Bob με κενές λίστες για τα ανταλλάξιμα bits, τη πόλωση και τα bits που κατάφεραν να περάσουν (για την τελευταία “0” ορίζεται η αποτυχία και “1” η

επιτυχία). Με βάση τις λίστες δημιουργούμε συνάρτηση για reset και μία για δημιουργία bits:

```
class Bob:
    bits=[] #0,1
    polarization=[]
    real_bits=[] #δοθέντα bits
    exchange=[]#Ανταλλάξιμα bits
    passed=[]#0:αποτυχία, 1:επιτυχία (τα bits που "περασαν")

    def reset(self):#δημιούργησε μία reset συνάρτηση για να είναι ευκολότερο να μετράει πολλές φορές
        self.bits=[]
        self.polarization=[]
        self.real_bits=[]
        self.exchange=[]
        self.passed=[]

    def create_bits(self, num_bits): #το num_bits αποτελεί είσοδο από την main
        for i in range(0,num_bits,1):
            x=random.randint(0,1)
            self.bits.append(x)#τα bits δημιουργούνται τυχαία και είναι τόσα όσο το πληθος των bits που στέλνονται
            if x==0 :
                self.polarization.append(-45)
            elif x==1:
                self.polarization.append(90)
        #print("bits παραγόμενα από Bob : ",self.bits)
        #print("πόλωση του Bob : ",self.polarization)
```

Στη συνέχεια δημιουργία συναρτήσεων, αρχικά για τα bits που θα λαμβάνει, και έπειτα για τη μέτρηση των «ανταλλάξιμων» για τη μέτρηση των QBER (πόσα πέρασαν ως είχαν και πόσα άλλαξαν) :

```
def measure_bits(self,received):#Λαμβάνει τη πόλωση που στέλνεται από την Alice/Eve ως παράμετρος
    for i in range(0,len(self.bits),1):
        if abs(self.polarization[i]-received[i])==90 :
            self.passed.append(0)
        elif (abs(self.polarization[i]-received[i])==45 or abs(self.polarization[i]-received[i])==135) :
            self.passed.append(random.randint(0,1))
    #print("ένδειξη bits που πέρασαν: ",self.passed)

def measure_real_bits(self,bits):#ρυθμός σφάλματος
    for i in range(0,len(self.bits),1):
        if self.passed[i]==1:#αν πέρασε
            self.real_bits.append(self.bits[i])#πρώτα πέρνα το bit για να ελεγχθεί(ανταλλάξιμο bit)
            if bits[i]==self.bits[i]:#Το όντως ανταλλάξιμο bit πρέπει να είναι ίδιο με το bit της Alice
                self.exchange.append(self.bits[i])
    #print("ανταλλάξιμα bits : ",self.real_bits)
    #print("πραγματικά ανταλλάξιμα bits : ",self.exchange)
    #print("πραγματικά ανταλλάξιμα bits/αυθεντικός αριθμός bits",len(self.exchange)/len(bits))
    #return len(self.exchange)/len(bits)
    print("QBER(Quantum Bit Error Rate) : ",(len(self.real_bits)-len(self.exchange))/len(self.real_bits)*100,"%")
    return (len(self.real_bits)-len(self.exchange))/len(self.real_bits)*100
```

Η τελευταία κλάση που δημιουργείται είναι αυτή για την Eve κατά την οποία επίσης δημιουργείται κενή λίστα για bits επιτυχίας και αποτυχίας, και συναρτήσεις για reset και δημιουργίας bits:

```
class Eve:
    polarization=[] #0,90,45,-45
    passed=[]#0:αποτυχία, 1:επιτυχία (bits που "πέρασαν")
    a_polarization=[]
    num_b=0

    def reset(self):#δημιούργησε μία reset συνάρτηση για να είναι ευκολότερο να μετράει πολλές φορές
        self.polarization=[]
        self.passed=[]
        self.a_polarization=[]

    def create_bits(self, num_bits): #το num_bits αποτελεί είσοδο από την main
        self.num_b=num_bits
        for i in range(0,num_bits,1):
            x=random.randint(0,1)
            if x==0 :
                self.polarization.append(-45)
            elif x==1:
                self.polarization.append(90)
        #print("πόλωση της Eve: ",self.polarization)
```

Και έπειτα συνάρτηση για την μέτρηση των bits με παρέμβαση ή μη του λαθρακροατή (της Eve):

```
def measure_bits(self,received,eavesdropping):#Λαμβάνει τη πόλωση που στέλνεται από την Alice ως παράμετρο
    for i in range(0,int(self.num_b*eavesdropping),1):#το σημείο όπου υπάρχει λαθρακουστής
        if abs(self.polarization[i]-received[i])==90 :
            self.passed.append(0)
        elif (abs(self.polarization[i]-received[i])==45 or abs(self.polarization[i]-received[i])==135) :
            self.passed.append(random.randint(0,1))

    for i in range(0,int(self.num_b*eavesdropping),1):
        if self.passed[i]==1:
            self.polarization[i]=45
        elif self.passed[i]==0:
            x=random.randint(0,1)
            if x==0:
                self.polarization[i]=0
            elif x==1:
                self.polarization[i]=45

    for i in range(int(self.num_b*eavesdropping),self.num_b,1):#μέρος χωρίς λαθρακουστή
        self.polarization[i]=self.a_polarization[i]
        self.passed.append(1)

    #print("πόλωση που στέλνεται πίσω από την Eve : ",self.polarization)
    #print("ένδειξη περασμένων bits : ",self.passed)


def send_polarization(self): #αποστολή στο κανάλι
    return self.polarization

def receive_p(self,alice):#Λήψη πόλωσης Alice
    self.a_polarization=alice
```

Όπως και με το BB84 θα χρειαστούμε τις βιβλιοθήκες pandas και matplotlib και ξεκινάμε με τη δημιουργία όμοιων λιστών και παραμέτρων:

```
from matplotlib import pyplot as plt
import pandas as pd

alice=Alice()
bob=Bob()
eve=Eve()
num_bits=[500]
bits=[]
basis=[]
qc_polarization=[]#πόλωση που στέλνεται από το κβαντικό κανάλι
count=0
QBER=[]
sum_QBER=0
sum_data=0
eavesdropping_rate=0.1
```

Επίσης η διαδικασία που ακολουθείται με απώτερο σκοπό τη μέτρηση του κβαντικού σφάλματος σε κάθε επίπεδο ρυθμού λαθρακρόασης είναι όμοια:

```
for i in range(0,len(num_bits)) :

    #print("αριθμος bits : ",num_bits[i])

    while(eavesdropping_rate<=1):

            random.seed()

            alice.reset()
            bob.reset()
            eve.reset()

            #int(input("πόσα bits να δημιουργηθούν? "))

            alice.create_bits(num_bits[i]) #Η Alice παράγει bits και τυχαία βάση
            qc_polarization=alice.send_polarization()
            eve.receive_p(qc_polarization)#μέρος όπου δεν υπάρχει λαθρακουστής με την πόλωση της Alice
            eve.create_bits(num_bits[i]) #Η Eve επιλέγει τη βάση τυχαία .
            eve.measure_bits(qc_polarization,eavesdropping_rate) #μέτρηση πόλωσης στη βάση της Eve
            qc_polarization=eve.send_polarization() #ετάδοση του αλλαγμένου σήματος στο κβαντικό κανάλι

            bob.create_bits(num_bits[i]) #Ο Bob τυχαία επιλέγει τη βάση
            bob.measure_bits(qc_polarization) #bέτρηση bits με βάση και πόλωση

            bits=alice.send_bits() #Η Alice στέλνει bits

            QBER.append(bob.measure_real_bits(bits)) #Ελέγχει αν είναι ίδια η βάση με της Alice και αποθηκεύει το πραγματικό bit

            eavesdropping_rate+=0.1
```

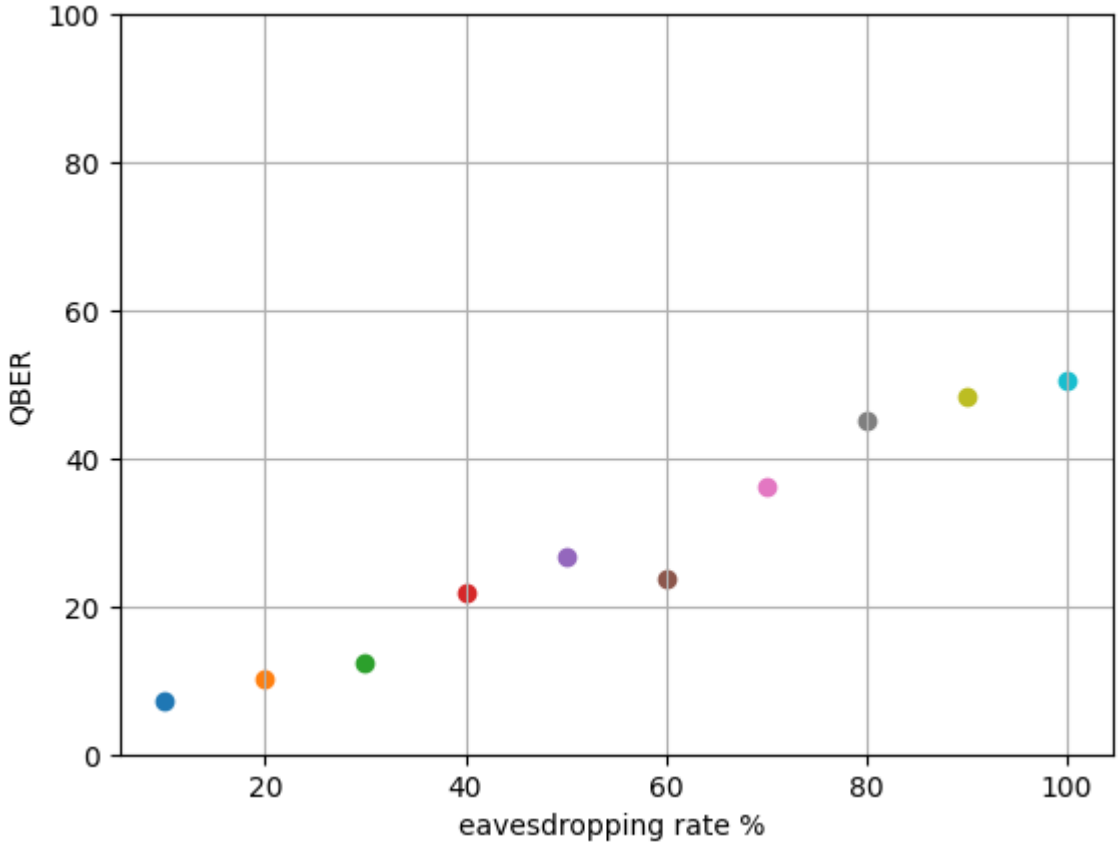
Όπως ακριβώς και για την οπτικοποίηση των αποτελεσμάτων:

```
z0=pd.DataFrame({'x0' : 10,'y0' : QBER[0]},index=[0])
z1=pd.DataFrame({'x1' : 20,'y1' : QBER[1]},index=[1])
z2=pd.DataFrame({'x2' : 30,'y2' : QBER[2]},index=[2])
z3=pd.DataFrame({'x3' : 40,'y3' : QBER[3]},index=[3])
z4=pd.DataFrame({'x4' : 50,'y4' : QBER[4]},index=[4])
z5=pd.DataFrame({'x5' : 60,'y5' : QBER[5]},index=[5])
z6=pd.DataFrame({'x6' : 70,'y6' : QBER[6]},index=[6])
z7=pd.DataFrame({'x7' : 80,'y7' : QBER[7]},index=[7])
z8=pd.DataFrame({'x8' : 90,'y8' : QBER[8]},index=[8])
z9=pd.DataFrame({'x9' : 100,'y9' : QBER[9]},index=[9])

plt.scatter(z0['x0'],z0['y0'])
plt.scatter(z1['x1'],z1['y1'])
plt.scatter(z2['x2'],z2['y2'])
plt.scatter(z3['x3'],z3['y3'])
plt.scatter(z4['x4'],z4['y4'])
plt.scatter(z5['x5'],z5['y5'])
plt.scatter(z6['x6'],z6['y6'])
plt.scatter(z7['x7'],z7['y7'])
plt.scatter(z8['x8'],z8['y8'])
plt.scatter(z9['x9'],z9['y9'])

plt.xlabel('eavesdropping rate % ')
plt.ylabel('QBER')
plt.ylim([0,100])
plt.grid()
plt.show()
```

Όπου πρώτα τυπώνονται οι τιμές του QBER για ρυθμό λαθρακρόασης από 10% μέχρι 100% με αύξηση κατά 10% για κάθε επανάληψη:

```
QBER(Quantum Bit Error Rate) :  7.352941176470589 %
QBER(Quantum Bit Error Rate) :  10.16949152542373 %
QBER(Quantum Bit Error Rate) :  12.4031007751938 %
QBER(Quantum Bit Error Rate) :  21.951219512195124 %
QBER(Quantum Bit Error Rate) :  26.666666666666668 %
QBER(Quantum Bit Error Rate) :  23.846153846153847 %
QBER(Quantum Bit Error Rate) :  36.283185840707965 %
QBER(Quantum Bit Error Rate) :  45.13888888888889 %
QBER(Quantum Bit Error Rate) :  48.30508474576271 %
QBER(Quantum Bit Error Rate) :  50.476190476190474 %
```

Και στην συνέχεια τα οπτικοποιούμε[31]:



4.6 Πρωτόκολλο E91

Το E91 πρωτόκολλο είναι γνωστό και ως EPR πρωτόκολλο ή πρωτόκολλο Ekert 91 καθώς δημιουργήθηκε από τον Artur Ekert το 1991 . Αποτελεί επίσης ένα ευρύτατα διαδεδομένο πρωτόκολλο και η κύρια διαφορά του με τα πρωτόκολλα που είδαμε προηγουμένως αποτελεί το γεγονός ότι βασίζεται στις ιδιότητες τις κβαντικής διεμπλοκής και της υπέρθεσης των σωματιδίων. Σε αυτό το πρωτόκολλο τα πολωμένα φωτόνια που βρίσκονται σε υπερθέσεις των 0 και 1, υπάρχουν ως ένα διαπλεγμένο ζεύγος στη μέση της Alice και του Bob. Με αυτόν τον τρόπο, δεν ειναι η Alice που στέλνει τα φωτόνια στον Bob, αλλά και οι δύο θα λάβουν από ένα φωτόνιο ο καθένας από τη πηγή που βρίσκεται στη μέση. Αυτό το πρωτόκολλο θα μπορούσε να είναι το κλειδί για την εφαρμογή μεγάλων αποστάσεων κβαντικής επικοινωνίας. Οι δορυφόροι θα μπορούσαν να θεωρηθούν ως ένα παράδειγμα πηγής στη μέση επικοινωνία. Αυτό αποτελεί και τη

βάση για να λειτουργήσει σωστά αυτό το πρωτόκολλο. Επίσης, αυτό το πρωτόκολλο είναι πολύ ανθεκτικό στην υποκλοπή, ακόμα και αν η πηγή έχει τον έλεγχο του λαθρακροατή λόγω της απλής ιδιότητας της κβαντικής διεμπλοκής. Ο λαθρακουστής δε μπορεί να προσδιορίσει τη κβαντική κατάσταση ενός σωματιδίου μέχρι να γνωρίσει τη κβαντική κατάσταση του άλλου που είναι πολύ μακριά από αυτό, γιατί τα σωματίδια συνδέονται μεταξύ τους[36].

4.7 Υλοποίηση E91

Για την υλοποίηση του E91 θα εγκαταστήσουμε αρχικά το "pycryptodome" (πακέτο για low-level κρυπτογραφία) και το "qit" ευρέως διαδεδομένο πακέτο για εφαρμογές κβαντικής πληροφορίας και κβαντικού υπολογισμού. Για την υλοποίηση αξιοποιήθηκε το [49] αποθετήριο github:

```
!pip install pycryptodome

Collecting pycryptodome
  Downloading pycryptodome-3.19.0-cp35-abi3-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (2.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.1/2.1 MB 10.8 MB/s eta 0:00:00
Installing collected packages: pycryptodome
Successfully installed pycryptodome-3.19.0
```

```
!pip install qit

Collecting qit
  Downloading qit-0.12.0-py3-none-any.whl (105 kB)
```

Έπειτα κάνουμε import κάποιες βασικές βιβλιοθήκες με την "CryptoRandom" να χρησιμοποιείται ως γεννήτρια παραγωγής ψευδοτυχαίων bits. Η πρώτη συνάρτηση που κατασκευάζεται είναι η bitFormat η οποία λαμβάνει μια λίστα με τιμές Boole (αντιπροσωπεύουν bits) ως είσοδο, και επιστρέφει μια εκτυπώσιμη αναπαράσταση αυτών των bits. Ελέγχει αν η λίστα είναι μικρότερη από το μέγιστο μέγεθος στο οποίο έχουμε αποδώσει μία τιμή (το 56), και αν ειναι, επιστρέφει μια μορφοποιημένη συμβολοσειρά των bits. Διαφορετικά, τα μετατρέπει σε δυαδική συμβολοσειρά, και στη συνέχεια σε δεκαεξαδική για εκτύπωση. Η επόμενη συνάρτηση που κατασκευάζεται είναι η "detectEavesdrop " που θα έχει ως είσοδο δύο λίστες bits που θα αντιπροσωπεύουν τα κλειδιά και τον ρυθμό σφάλματος. Θα ελέγχει αν κάποιο εκ των κλειδιών είναι κενό ή διαφέρουν τα μήκη τους (ένδειξη πιθανής λαθρακρόασης). Υπολογίζει την αναντιστοιχία μεταξύ των δύο κλειδιών και ελέγχει αν το ποσοστό αναντιστοιχίας υπερβαίνει ένα όριο με βάση τον ρυθμό σφάλματος.

```
import numpy as np
from Crypto.Random import random
import qit

MAX_PRINT_SIZE = 56

def bitFormat(bits):
    """Return a printable representation of the given list of bools representing bits."""
    if len(bits) < MAX_PRINT_SIZE:
        out = ', '.join(['1' if b == True else '0' if b == False else '-' for b in bits])
        return '[' + out + ']'
    else:
        binaryString = ""
        for j in range(len(bits)):
            if bits[j]:
                binaryString += "1"
            else: binaryString += "0"
        return hex(int(binaryString, 2))

def detectEavesdrop(key1, key2, errorRate):
    """Return True if Alice and Bob detect Eve's interference, False otherwise."""
    if len(key1) == 0 or len(key2) == 0:
        return True
    if len(key1) != len(key2):
        return True

    tolerance = errorRate * 1.2
    mismatch = sum([1 for k in range(len(key1)) if key1[k] != key2[k]])
    if abs((float(mismatch) / len(key1)) - errorRate) > tolerance:
        return True

    return False
```

Έπειτα με την "discloseHalf" διαιρεί κάθε κλειδί σε δύο μέρη (ένα που ανακοινώνει και ένα που κρατάει-τα εναπομείναντα bits), την "equivState" για έλεγχο αν οι δύο καταστάσεις αντιπροσωπεύουν την ίδια κβαντική κατάσταση συγκρίνοντας τις πιθανότητές τους.

```
def discloseHalf(key1, key2):
    """Return the tuple (announce1, keep1, announce2, keep2), where
           announce2, announce2 = bit values to announce and discard
           keep1, keep2 = bit values of new shared keys
    """
    # Disclose every other bit
    announce1 = [key1[j] for j in range(len(key1)) if j % 2 == 0]
    keep1 = [key1[j] for j in range(len(key1)) if j % 2]
    announce2 = [key2[j] for j in range(len(key2)) if j % 2 == 0]
    keep2 = [key2[j] for j in range(len(key2)) if j % 2]
    return (announce1, keep1, announce2, keep2)

def equivState(state1, state2):
    """Return True if state1 and state2 represent the same quantum state."""
    return np.array_equal(state1.prob(), state2.prob())

def getRandomBits(length):
    """Return a list of bits with given length, each either 0 or 1 with equal probability."""
    bitstring = []
    for j in range(length):
        bitstring.append(random.choice([True, False]))
    return bitstring
```

Στη συνέχεια ξεκινάει η μοντελοποίηση της λειτουργίας του E91. Θα επιλεχθούν τυχαίοι άξονες μέτρησης για την Alice και τον Bob για συγκεκριμένα αριθμό qubits:

```
def chooseAxes(numBits):
    """Return Alice and Bob's randomly chosen mstment axes for the specified
       number of qubits in the E91 protocol:
           A chooses from (0, pi/4, pi/2) with equal probability,
           B chooses from (pi/4, pi/2, 3pi/4) with equal probability.
    """
    choicesA = [0, pi/8, pi/4]
    choicesB = [0, pi/8, -pi/8]
    basesA = []
    basesB = []
    for j in range(numBits):
        basesA.append(random.choice(choicesA))
        basesB.append(random.choice(choicesB))

    return (basesA, basesB)

def formatBasesForPrint(bases):
    """Return printable representation of E91 basis choices for Alice and Bob.
           value | angle
             1   |   0
             2   |  pi/8
             3   |  pi/4
             4   | -pi/8
    """
    out = []
    for j in range(len(bases)):
        if bases[j] == 0:
            out.append(1)
        elif bases[j] == pi/8:
            out.append(2)
        elif bases[j] == pi/4:
            out.append(3)
        elif bases[j] == -pi/8:
            out.append(4)

    return out
```

Δοθέντων των αποτελεσμάτων και των βάσεων μέτρησης, θα αφαιρεθούν τα bits για τα οποία η Alice και ο Bob διάλεξαν ασυμβίβαστες βάσεις. Έπειτα θα γίνει προσομοίωση της μέτρησης των qubits σε κατάσταση διεμπλοκής βασισμένων στις βάσεις της Alice και του Bob, ενώ ο ρυθμός σφάλματος δείχνει πιθανά σφάλματα μέτρησης:

```
def matchKeys(key1, key2, bases1, bases2):
    """Return the tuple (key1, key2, discard1, discard2) after removing bits where Alice
    and Bob selected incompatible axes of measurement in the E91 protocol.
    """
    match = [True if bases1[k] == bases2[k] else False for k in range(len(bases1))]
    discard1 = [key1[k] for k in range(len(key1)) if not(match[k])]
    key1 = [key1[k] for k in range(len(key1)) if match[k]]
    discard2 = [key2[k] for k in range(len(key2)) if not(match[k])]
    key2 = [not(key2[k]) for k in range(len(key2)) if match[k]]

    return (key1, key2, discard1, discard2)

def measureEntangledState(basisA, basisB, errorRate=0.0):
    """Return Alice and Bob's measurement results on a pair of maximally
    entangled qubits. basis[A,B] contain Alice and Bob's axes of mstment.
    """
    # Alice measures either basis state with equal probability
    # -1 will correspond to False (0) and +1 will correspond to True (1)
    resultA = random.choice([-1, 1])

    # If Alice and Bob chose the same axis of mstment, Bob's result is
    # perfectly anti-correlated with Alice's. Otherwise its correlation
    # coefficient is given by -cos[2(basisA-basisB)]. We use the result
    # r to generate a correlated random number that gives Bob's result.
    r = -1 * cos(2 * (basisA - basisB))
    r2 = r ** 2
    ve = 1 - r2
    SD = sqrt(ve)
    e = np.random.normal(0, SD)
    resultB = resultA * r + e

    resultA = False if resultA < 0 else True
    resultB = False if resultB < 0 else True

    if errorRate:
        samples = np.random.rand(2)
        if samples[0] < errorRate: resultA = not(resultA)
        if samples[1] < errorRate: resultB = not(resultB)

    return (resultA, resultB)
```

Συνεπώς τώρα μπορεί να δομηθεί η συνάρτηση που θα χρησιμοποιηθεί αργότερα για την πρακτική εφαρμογή του E91.

```
def runE91(n, errorRate=0.0, verbose=True):
    """Simulation of Ekert's 1991 entanglement-based protocol for quantum key distribution."""
    numBits = 5 * n

    if verbose:
        print("\n=====E91 protocol=====\n%d initial bits, ~%d key bits" % (numBits, n))
        print("without eavesdropping")
        if errorRate: print("with channel noise\n")
        else: print("without channel noise\n")

    # A trusted mediator generates pairs of particles in the singlet state
    #     +0.7071 |0> -0.7071 |1>
    # and sends one particle from each pair to Alice and the other to Bob.
    # Alice randomly offsets her axis of measurement by one of the following:
    #     [0, pi/8, pi/4]
    # Bob randomly offsets his axis of measurement by one of the following:
    #     [0, pi/8, -pi/8]
    bases_A, bases_B = e91.chooseAxes(numBits)
    key_A, key_B = [], []

    for j in range(numBits):
        (new_A, new_B) = e91.measureEntangledState(bases_A[j], bases_B[j], errorRate)
        key_A.append(new_A)
        key_B.append(new_B)

    print("Alice's randomly chosen axes of measurement:\n%s" % e91.formatBasesForPrint(bases_A)
    print("Bob's randomly chosen axes of measurement:\n%s" % e91.formatBasesForPrint(bases_B))
    print("Alice's measurement results:\n%s" % util.bitFormat(key_A))
    print("Bob's measurement results:\n%s" % util.bitFormat(key_B))

    key_A, key_B, discard_A, discard_B = e91.matchKeys(key_A, key_B, bases_A, bases_B)
    print("Alice's %d discarded bits:\n%s" % (len(discard_A), util.bitFormat(discard_A)))
    print("Bob's %d discarded bits:\n%s" % (len(discard_B), util.bitFormat(discard_B)))

    print("Alice's %d-bit sifted key:\n%s" % (len(key_A), util.bitFormat(key_A)))
    print("Bob's %d-bit sifted key:\n%s" % (len(key_B), util.bitFormat(key_B)))

    return key_A
```

Για την πρακτική υλοποίηση αρχικά θα κατασκευαστεί η κλάση “Alice” η οποία θα επιστρέφει μία λίστα διαμπλεκόμενων ζευγαριών bits, που θα αναπαρίστανται ως πλειάδες των δύο τυχαίων δυαδικών τιμών. Η κλάση “Bob” που θα κατασκευαστεί στη συνέχεια, προσομοιώνει τη μέτρηση του Bob για τα διαμπλεκόμενα ζευγάρια bits, καθώς θα τα δέχεται από την Alice ως είσοδο και θα διαλέγει ένα bit από κάθε ζευγάρι. Τέλος θα κατασκευαστεί και μία κλάση για το E91 ώστε να υλοποιηθούν τα βήματα της διαδικασίας του με την αντίστοιχη εκτύπωση του QBER από τη σύγκριση των αποτελεσμάτων μέτρησης της Alice και του Bob:

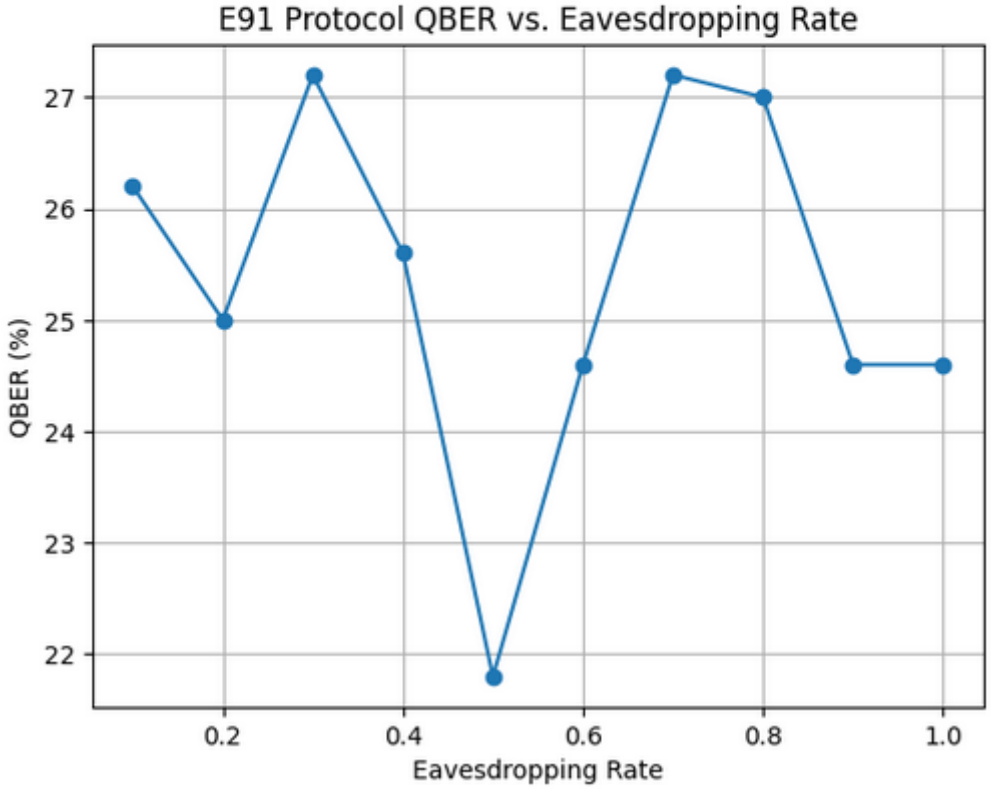
```python
import random
from matplotlib import pyplot as plt
import pandas as pd

class Alice:
    def create_entangled_pairs(self, num_pairs):
        entangled_pairs = [(random.randint(0, 1), random.randint(0, 1)) for _ in range(num_pairs)]
        return entangled_pairs

class Bob:
    def measure_entangled_pairs(self, entangled_pairs):
        measured_bits = [random.choice(pair) for pair in entangled_pairs]
        return measured_bits

class E91Protocol:
    def run_protocol(self, num_pairs, eavesdropping_rate):
        alice = Alice()
        bob = Bob()
        entangled_pairs = alice.create_entangled_pairs(num_pairs)

        # Alice and Bob perform measurements on their entangled particles
        alice_bits = [pair[0] for pair in entangled_pairs]
        bob_bits = bob.measure_entangled_pairs(entangled_pairs)

        # Calculate QBER
        qber = sum([1 for a, b in zip(alice_bits, bob_bits) if a != b]) / num_pairs * 100
        return qber
```

Τέλος, γίνεται η υλοποίηση αφού πρώτα οριστούν οι παράμετροι, δηλαδή 500 διαμπλεκόμενα bits και ρυθμός λαθρακρόασης από το 0.1 έως το 1 με 0.1 ρυθμό αύξησης σε κάθε επανάληψη:

```python
def main():
    num_pairs = 500
    eavesdropping_rates = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
    qber_results = []

    e91_protocol = E91Protocol()

    for eavesdropping_rate in eavesdropping_rates:
        qber = e91_protocol.run_protocol(num_pairs, eavesdropping_rate)
        qber_results.append(qber)
        print(f"QBER for eavesdropping rate {eavesdropping_rate}: {qber:.2f}%")

    plt.plot(eavesdropping_rates, qber_results, marker='o')
    plt.xlabel('Eavesdropping Rate')
    plt.ylabel('QBER (%)')
    plt.title('E91 Protocol QBER vs. Eavesdropping Rate')
    plt.grid()
    plt.show()

if __name__ == "__main__":
    main()
```

Εμφανίζονται τα αποτελέσματα των QBER για κάθε ρυθμό λαθρακρόασης τόσο τυπωμένα όσο και σε διάγραμμα:

```
QBER for eavesdropping rate 0.1: 26.20%
QBER for eavesdropping rate 0.2: 25.00%
QBER for eavesdropping rate 0.3: 27.20%
QBER for eavesdropping rate 0.4: 25.60%
QBER for eavesdropping rate 0.5: 21.80%
QBER for eavesdropping rate 0.6: 24.60%
QBER for eavesdropping rate 0.7: 27.20%
QBER for eavesdropping rate 0.8: 27.00%
QBER for eavesdropping rate 0.9: 24.60%
QBER for eavesdropping rate 1.0: 24.60%
```



Τώρα, απαραίτητο κρίνεται να γίνει επανάληψη της προηγούμενης υλοποίησης αλλά με την επαύξηση του κώδικα με εισαγωγή της κλάσης της Eve, ώστε να προσομοιωθεί η λαθρακρόαση την οποία επιφέρει:

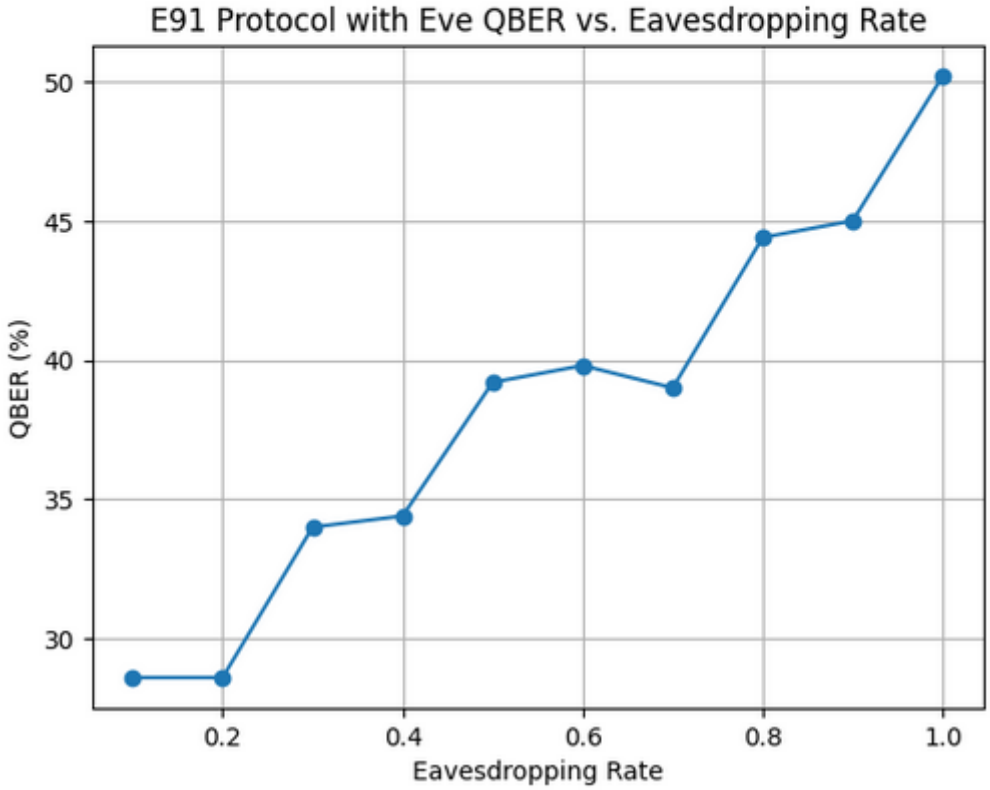
```
class Eve:
    def eavesdrop(self, entangled_pairs, eavesdropping_rate):
        eavesdropped_pairs = []
        for a, b in entangled_pairs:
            if random.random() <= eavesdropping_rate:
                # Eve's eavesdropping action
                eavesdropped_pairs.append((random.randint(0, 1), random.randint(0, 1)))
            else:
                eavesdropped_pairs.append((a, b))
        return eavesdropped_pairs
```

Και στη συνέχεια ανανεώνεται ο κώδικας με την παρουσία της Eve στη διαδικασία του E91 με τον ρόλο του υποκλοπέα:

```
class E91ProtocolWithEve:
    def run_protocol(self, num_pairs, eavesdropping_rate):
        alice = Alice()
        bob = Bob()
        eve = Eve()

        entangled_pairs = alice.create_entangled_pairs(num_pairs)
        eavesdropped_pairs = eve.eavesdrop(entangled_pairs, eavesdropping_rate)

        alice_bits = [pair[0] for pair in entangled_pairs]
        bob_bits = bob.measure_entangled_pairs(eavesdropped_pairs)

        qber = sum([1 for a, b in zip(alice_bits, bob_bits) if a != b]) / num_pairs * 100
        return qber

def main():
    num_pairs = 500
    eavesdropping_rates = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
    qber_results = []

    e91_protocol_with_eve = E91ProtocolWithEve()

    for eavesdropping_rate in eavesdropping_rates:
        qber = e91_protocol_with_eve.run_protocol(num_pairs, eavesdropping_rate)
        qber_results.append(qber)
        print(f"QBER for eavesdropping rate {eavesdropping_rate}: {qber:.2f}%")

    plt.plot(eavesdropping_rates, qber_results, marker='o')
    plt.xlabel('Eavesdropping Rate')
    plt.ylabel('QBER (%)')
    plt.title('E91 Protocol with Eve QBER vs. Eavesdropping Rate')
    plt.grid()
    plt.show()

if __name__ == "__main__":
    main()
```

Και με αντίστοιχο τρόπο όπως και προηγουμένως τυπώνονται τα αποτελέσματα για το QBER καθώς και αποτυπώνονται σε γράφημα[49]:

```
QBER for eavesdropping rate 0.1: 28.60%
QBER for eavesdropping rate 0.2: 28.60%
QBER for eavesdropping rate 0.3: 34.00%
QBER for eavesdropping rate 0.4: 34.40%
QBER for eavesdropping rate 0.5: 39.20%
QBER for eavesdropping rate 0.6: 39.80%
QBER for eavesdropping rate 0.7: 39.00%
QBER for eavesdropping rate 0.8: 44.40%
QBER for eavesdropping rate 0.9: 45.00%
QBER for eavesdropping rate 1.0: 50.20%
```



4.8 Συμπεράσματα

Μεταξύ της πληθώρας πρωτοκόλλων διανομής κβαντικού κλειδιου, τα BB84 και B92 αποτελούν αυτά που αποτέλεσαν αντικείμενο τριβής προηγουμένως και αποτελούν κάποια από

τα πιο συχνά στη βιβλιογραφία. Παρά το γεγονός ότι έχουν πολλά στοιχεία κοινά προκύπτουν κάποιες σημαντικές διαφορές μεταξύ τους. Η κυριότερη που ήδη έχει αναφερθεί αποτελεί το γεγονός ότι το BB84 χρησιμοποιεί τέσσερις καταστάσεις ενώ το B92 χρησιμοποιεί δύο. Ένα άμεσο θετικό του B92 αποτελεί το γεγονός ότι οι δύο καταστάσεις του παρέχουν απλότητα στην εφαρμογή του πρωτοκόλλου καθώς διαθέτει λιγότερο υπολογιστικό κόστος. Ωστόσο, γενικά μιλώντας, για τον ίδιο λόγο η απόδοσή του δεν είναι εξίσου καλή με του BB84. Η παρουσία μόλις δύο γραμμικά ανεξάρτητων καταστάσεων καθιστούν ευκολότερο το έργο του λαθρακροατή να εκτελέσει μετρήσει επί των κβαντικών καταστάσεων που ετοιμάζονται από την Alice[35]. Αυτό έγινε αντιληπτό και από την εφαρμογή και των δύο πρωτοκόλλων καθώς σε κάθε επίπεδο λαθρακρόασης το κβαντικό σφάλμα ήταν αισθητά μεγαλύτερο στο B92 συγκριτικά με το BB84, με αποτέλεσμα στο μέγιστο ποσοστό λαθρακρόασης το πρώτο να έχει QBER τάξης του 50% ενώ το δεύτερο τάξης περίπου 25%, μόλις το μισό. Γενικά, το BB84 παρέχει άνευ όρων ασφάλεια, που σημαίνει ότι είναι ασφαλές έναντι οποιασδήποτε επίθεσης υποκλοπής, ενώ το B92 παρέχει υπό προϋποθέσεις, δηλαδή ικανοποιείται υπό τήρηση κάποια συγκεκριμένης συνθήκης (συγκεκριμένα αν ο υποκλοπέας δεν κάνει καμία μέτρηση στο κβαντικό κανάλι). Ωστόσο, το BB84 είναι το πιο ευρέως χρησιμοποιούμενο και μελετημένο, το οποίο γίνεται αντιληπτό και από τα αποτελέσματα που δίνει, που ως απόρροια αυτών υπάρχει πληθώρα εμπορικών διαθέσιμων συστημάτων που το υλοποιούν. Από την άλλη το B92 αποτελεί νεότερο πρωτόκολλο που δεν έχει εφαρμοστεί σε εμπορικό σύστημα. Ακόμη και θεωρητικά , το πρωτόκολλο B92 δεν είναι τόσο ασφαλές όσο άλλοι σύγχρονοι μέθοδοι κρυπτογράφησης, και πιθανότατα να πρέπει να αναπτυχθεί και να βελτιστοποιηθεί προτού αντικατασταθούν τελικά τα σημερινά συστήματα κρυπτογράφησης. Εκτός αν οι τρέχουσες κρυπτογραφικές προσεγγίσεις καταστούν απαρχαιωμένες κατά κάποιο τρόπο, το B92 δε φαίνεται αρκετά βιώσιμη αντικατάσταση μέχρι να ερευνηθεί και βελτιστοποιηθεί περαιτέρω[34]. Συνοπτικά, το BB84 παρέχει άνευ όρων ασφάλεια με πολύ λιγότερα κβαντικά σφάλματα αλλά απαιτεί περισσότερη υπολογιστική πολυπλοκότητα έναντι του B92 που έχει ασφάλεια υπό όρους αλλά απαιτεί λιγότερους υπολογιστικούς πόρους. Όσο αφορά το E91, το QBER έμεινε σχετικά σταθερό στη περίπτωση που η Eve δεν παρεμβαίνει, ενώ όταν παρεμβαίνει το QBER διαρκώς αυξάνεται, φτάνοντας σε σχεδόν διπλάσιο σφάλμα μετά το πέρας των υλοποιήσεων για κάθε ρυθμό λαθρακρόασης. Αυτό υποδεικνύει την ευπάθεια του συστήματος στην υποκλοπή. Ωστόσο, γενικά το E91 παρέχει απλότητα σχετικά με την υλοποίηση. Η επιλογή μεταξύ αυτών των πρωτοκόλλων εξαρτάται από τις συγκεκριμένες απαιτήσεις του συστήματος επικοινωνίας και τους διαθέσιμους πόρους προς υλοποίηση.

4.9 Υλοποιήσεις σε πραγματικά περιβάλλοντα

- Το 2007 το Los Alamos National Laboratory/NIST πέτυχε διανομή κβαντικού κλειδιού σε 148.7 km οπτικής ίνας χρησιμοποιώντας το πρωτόκολλο BB84. Είναι σημαντικό ότι η απόσταση αυτή είναι αρκετά μεγάλη για σχεδόν όλα τα ανοίγματα που βρίσκονται στα σημερινά δίκτυα οπτικών ινών[29].

- Το 2008, η ανταλλαγή κλειδιών ασφαλείας σε 1 Mbit/s (πάνω από 20km οπτικής ίνας) και 10 kbits/s (πάνω από 20km οπτικής ίνας), επιτεύχθηκε με συνεργασία μεταξύ του Πανεπιστημίου του Cambridge και της Toshiba χρησιμοποιώντας το πρωτόκολλο BB84[29].

- Τον Αύγουστου του 2015 η μεγαλύτερη απόσταση για οπτική ίνα (307 km) επιτεύχθηκε από το University of Geneva and Corning Inc. Στο ίδιο πείραμα δημιουργήθηκε ένας μυστικός ρυθμός κλειδιού 12.7 kbit/s, καθιστώντας τον υψηλότερο ρυθμό bit σε αποστάσεις των 100 km[29].

- Τον Ιούνιο του 2017 φυσικοί με επικεφαλείς τον Thomas Jennewein στο Institute for Quantum Computing και στο πανεπιστήμιο του Waterloo στο Waterloo του Καναδά, πέτυχαν τη πρώτη επίδειξη κατανομής κβαντικού κλειδιού από έναν επίγειο πομπό σε ένα κινούμενο αεροσκάφος. Ανάφεραν οπτικές συνδέσεις με αποστάσεις μεταξύ των 3-10km και δημιούργησαν ασφαλή κλειδιά μήκους έως 868 kilobyte[29].

- Τον Μάϊο του 2019, μια ομάδα με επικεφαλή τον Hong Guo στο Πανεπιστήμιο του Πεκίνου και στο πανεπιστήμιο των ταχυδρομείων και τηλεπικοινωνιών του Πεκίνου, ανάφερε δοκιμές πεδίου ενός συστήματος QKD συνεχούς μεταβλητής μέσω εμπορικών οπτικών ινών στη Xi'an και Guangzhou σε αποστάσεις 30.02 km (12.48 dB) και 49.85 km (11.62 dB) αντίστοιχα[29].

- Το 2023, οι επιστήμονες του ινδικού ινστιτούτου τεχνολογίας (IIT) του Delhi πέτυχαν διανομή κβαντικού κλειδιού χωρίς αξιόπιστους κόμβους έως και 380 km σε τυπική τηλεπικοινωνιακή ίνα με πολύ χαμηλό ποσοστό σφαλμάτων κβαντικών bit (QBER)[29].

4.10 Άλλα πρωτόκολλα QKD

Στην προσπάθεια βελτίωσης προβλημάτων που προκυπτούν στα πρωτόκολλα QKD, όπως αυτά που παρουσιάστηκαν προηγουμένως, αναδύεται πληθώρα νέων:

- BBM92: αποτελεί QKD που αναπτύχθηκε με τη χρησιμοποίηση διαπλεγμένων πολωμένων ζευγών φωτονίων από τους Charles H. Bennett, Gilles Brassard and N. David Mermin το 1992 (στους οποίους οφείλει την ονομασία του). Χρησιμοποιεί μόνο δύο καταστάσεις αντί για 4 που χρησιμοποιούνται στα BB84 και E91. Χρησιμοποιείται για μη ορθογώνια κβαντική μετάδοση. Το 0 μπορεί να αποκρυπτογραφηθεί ως 0 μοίρες και το 1 ως 45 μοίρες σε διαγώνιο πρωτόκολλο BB92. Δεν υπάρχουν ασφαλείς υποκλοπές και ασφάλειες για απόσταση 200-300 μέτρα[37].

- Six-state protocol (SSP): Το πρωτόκολλο έξι καταστάσεων αποτελεί μία παραλλαγή του BB84 που χρησιμοποιεί έξι καταστάσεις σε τρεις ορθογώνιες βάσεις, και εμφανίστηκε το 1998 σε άρθρο του Dagmar Bruss. Αποτελεί πρωτόκολλο διακριτής μεταβλητής για QKD που επιτρέπει την ανοχή ενός πιο θορυβώδους καναλιού από το πρωτόκολλο BB84. Το SSP παράγει υψηλότερο ποσοστό σφαλματων κατά την απόπειρα λαθρακρόασης, οπότε το κάνει ευκολότερο να εντοπιστούν σφάλματα, καθώς ο υποκλοπέας πρέπει να διαλέξει μεταξύ τριών πιθανών βάσεων. Τα συστήματα υψηλών διαστάσεων έχουν αποδειχθεί ότι παρέχουν υψηλότερο επίπεδο ασφάλειας[38].

- SAR04: Το 2004 οι ερευνητές κατασκεύασαν το SAR04, όταν παρατήρησαν ότι χρησιμοποιώντας τις τέσσερις καταστάσεις του BB84 με διαφορετική κωδικοποίηση πληροφοριών θα μπορούσαν να αναπτύχουν ένα νέο πρωτόκολλο που θα ήταν πιο ισχυρό, ειδικά έναντι στην επίθεση διαχωρισμού των φωτονίων, όταν χρησιμοποιούνται εξασθενημένοι παλμοί λέιζερ αντί για απλές πηγές φωτονίων. Το πλεονέκτημα αυτού έναντι του απλούστερου BB84 είναι ότι η Alice δεν ανακοινώνει ποτέ τη βάση του bit της. Ως αποτέλεσμα, η Eve χρειάζεται να αποθηκεύσει περισσότερα αντίγραφα του qubit για να μπορέσει να προσδιορίσει τελικά τη κατάσταση από ο,τι θα έκανε αν η βάση ανακοινωνόταν απευθείας. Σε υλοποιήσεις ενός φωτονίου θεωρήθηκε ότι είναι ισοδύναμο με το BB84, αλλά τα πειράματα έχουν δείξει ότι είναι κατώτερο[39].

- COW πρωτόκολλο: Το COW (coherent one-way) αποτελεί ένα νέο πρωτόκολλο που αναπτύχθηκε το 2004. Είναι προσαρμοσμένο για να λειτουργεί με ασθενείς συνεκτικούς παλμούς σε υψηλούς ρυθμούς bit. Το πλεονέκτημα αυτού του συστήματος είναι ότι η εγκατάσταση είναι πειραματικά απλή και είναι ανεκτική σε μειωμένη ορατότητα

παρεμβολών και σε επιθέσεις διαχωρισμού αριθμών φωτονίων, με αποτέλεσμα υψηλή απόδοση όσο αφορά τα αποσταγμένα μυστικά bits ανά qubit[40].

- KMB09 πρωτόκολλο: Το πρωτόκολλο KMB09 αποτελεί ένα εναλλακτικό πρωτόκολλο QKD όπου χρησιμοποιούν η Alice και ο Bob δύο αμοιβαία αμερόληπτες βάσεις με τη μία να κωδικοποιεί το "0" και την άλλη το "1". Η ασφάλεια του συστήματος οφείλεται σε ενα ελάχιστον ποσοστό σφάλματος μετάδοσης δείκτη (ITER) και στο ποσοστό κβαντικού σφάλματος bit (QBER) που εισήγαγε ένας υποκλοπέας. Το ITER αυξάνεται σημαντικά για καταστάσεις φωτονίων υψηλότερων διαστάσεων. Αυτό επιτρέπει περισσότερο θόρυβο στη γραμμή μεταφοράς, αυξάνοντας έτσι πιθανή απόσταση μεταξύ της Alice και του Bob χωρίς να χρειάζονται ενδιάμεσοι κόμβοι[40].

- S09 πρωτόκολλο: Το κβαντικό πρωτόκολλο S09 βασίζεται σε κρυπτογραφία δημόσιου ιδιωτικού κλειδιού για ασφαλή μεταφορά δεδομένων μέσω δημόσιου καναλιού. Η ασφάλεια του πρωτοκόλλου απορρέει από το γεγονός ότι η Alice και ο Bob χρησιμοποιούν ο καθένας μυστικά κλειδιά σε πολλαπλή ανταλλαγή qubit, σε αντίθεση με το BB84 και τις παραλλαγές του. Ο Bob ξέρει το κλειδί για τη μετάδοση, τα qubits μεταδίδονται μόνο προς τη μία κατεύθυνση και ανταλλάσονται κλασικές πληροφορίες. Στη συνέχεια η επικοινωνία στο πρωτόκολλο παραμένει κβαντική σε κάθε στάδιο. Στο BB84 κάθε μεταδιδόμενο qubit βρίσκεται σε μία από τις τέσσερις διαφορετικές καταστάσεις. Σε αυτό το πρωτόκολλο το qubit που μεταδίδεται μπορεί να είναι σε οποιαδήποτε από τις αυθαίρετες καταστάσεις[40].

4.11 Κβαντική γεννήτρια τυχαίων αριθμών

Κατά την πληθώρα εργασιών που ανατίθενται τα τελευταία χρόνια στους υπολογιστές, αυτοί λειτουργούν με αιτιοκρατικό τρόπο απόκρισης προς επίλυση, δηλαδή θα παρέχουν κάθε φορά την ίδια απάντηση για την ίδια ερώτηση. Δηλαδή η φύση της κατασκευής τους είναι τέτοια ώστε να εξαλείφεται η τυχαιότητα κατά την εξαγωγή των αποτελεσμάτων. Ωστόσο, το ίδιο γεγονός μπορεί να αποτελεί πρόβλημα σε άλλες διαδικασίες που η τυχαιότητα είναι αναγκαία (π.χ. το πεδίο της κυβερνοασφάλειας).

Ο όρος τυχαιότητα χαρακτηρίζει την έλλειψη τάξης και οργάνωσης , και η αύξησή της είναι ανάλογη με την αύξηση εντροπίας του συστήματος. Αν βασίζεται στη τάξη και την οργάνωση (έστω και σε μικρό βαθμό) τότε της αποδίδουμε τον όρο ψευδοτυχαιότητα. Οι υπολογιστές βασίζονται σε αυτήν για τη δημιουργια ψευδοτυχαίων αριθμών μέσω γεννητριών που τις παράγουν. Ο σκοπός αυτών είναι η παραγωγή γρήγορων και φθηνών ανεξάρτητων bits με ομοιόμορφη κατανομή πιθανότητας. Έτσι η γνώση μερικών δυαδικών ψηφίων δε παρέχουν καμία ουσιαστική γνώση για τα υπόλοιπα, με τη συγκεκριμένη μέθοδο να βρίσκει εφαρμογή σε πληθώρα εφαρμογών, για παράδειγμα στην εγγύηση ασφάλειας των (ραγδαίως εξελισσόμενων) κρυπτονομισμάτων. Οι σημερινές γεννήτριες αριθμών εξαρτώνται από αλγορίθμους υπολογιστών, γεγονός που τις καθιστά αιτιοκρατικές (ντετερμινιστικές). Αν και οι αριθμοί που προκύπτουν φαίνονται να είναι πραγματικά τυχαίοι, στην ουσία δεν είναι παρά μόνο φαινομενική αυτή η τυχαιότητα, καθώς η γνώση ορισμένων πληροφοριών για την είσοδο, είναι εφικτό να δώσουν δικαίωμα πρόσβασης σε κάποιον τρίτο. Στην πραγματικότητα, κύριο στόχο αποτελεί η δημιουργία ακολουθίας δυαδικών ψηφίων, που δεν είναι απόλυτα τυχαίοι, και έπειτα να ανακατευτούν με την επίδραση μιας συνάρτησης προς παραγωγής τυχαίας σειράς bits. Συνεπώς αναδύεται η ανάγκη για όλα τα κρυπτοσυστήματα (κλασικά και κβαντικά) να εκμεταλλεύονται πραγματικά τυχαίους και απρόβλεπτους αριθμούς.

Υπό αυτά τα δεδομένα έρχονται τα τρέχοντα επίπεδα κβαντικής τεχνολογίας, τα οποία εκτός από τη διανομή κλειδιών, αποτελούν καλή πηγή γνήσιας τυχαιότητας, η οποία αποδείχθηκε εξαιρετικά σημαντική για την κρυπτογραφία και για τους αλγόριθμους προς προσομοίωση. Ως «γνήσια» τυχαιότητα αναφερόμαστε σε μια πηγή της οποίας η έξοδος είναι απρόβλετη και αδύνατο να παραχθεί βασιζόμενη σε γνωστούς φυσικούς νόμους. Έτσι αντιτίθεται στον ντετερμινιστικο χαρακτήρα παραγωγής συμβολοσειρών που παρέχουν οι ψευδοτυχαίες γεννήτριες αριθμών, που έχουν προκαθοριστεί με αιτιοκρατικό αλγόριθμο. Δηλαδή οι αριθμοί που παράγονται κατανέμονται αδιάκριτα από μία ομοιόμορφη κατανομή πιθανοτήτων. Η στιβαρότητα των γεννητριών ψευδοτυχαίων αριθμών, αποτελεί ζήτημα που χρήζει προσεκτικής εξέτασης.

Η πιο άμεση κβαντική γεννήτρια τυχαίων αριθμών εκμεταλλεύεται την τυχαιότητα των αποτελεσμάτων στις κβαντικές μετρήσεις, για παράδειγμα διαβάζοντας την έξοδο ενός διαχωριστή δέσμης 50/50. Ο τελευταίος αποτελεί σύστημα εκμετάλλευσης κβαντικών διακυμάνσεων κενού, καθώς τις κατευθύνει μέσω δέσμης laser σε συσκευή

που τις ανάγει σε αριθμούς. Η δέσμη κατευθύνεται σε διαχωριστή, κατευθύνει σε (προστατευμένο από εξωτερικές πηγή) διαχωριστή, ο οποίος επενεργεί επί αυτής διαχωρίζοντάς την. Οι δύο προκύπτουσες δέσμες καταλήγουν σε δύο ανιχνευτές όπου θα μετατραπούν σε ηλεκτρονικά σήματα. Η αποφυγή της επίδρασης κβαντικών διακυμάνσεων του κενού κατορθώνει την απομάκρυνση του θορύβου που πηγάζει από αυτόν, και ανάγεται σε σειρά πραγματικά τυχαίων αριθμών. Η ταχύτητα της δέσμης φτάνει τα 6.5 Mbps και μπορεί να αναμετρηθεί με εκείνες των εμπορικά διαθέσιμων γεννητριών.

Μία νέα μέθοδος τυχαιότητα διασφαλισμένης από τη κβαντική μηχανική που αναδύεται είναι αυτή των τσιπ, που βασίζονται στις (απρόβλεπτες) κβαντικές ιδιότητες του φωτός. Τα προαναφερθέντα τσιπ παράγουν και εν τέλει μία ακολουθία αυθεντικών τυχαίων αριθμών, που μπορούν να χρησιμοποιηθούν ως κλειδιά κρυπτογράφησης, τα οποία ανεξάρτητα από το διαθέσιμο πλήθος δεδομένων που μπορεί να έχει κανείς, δεν καθίσταται ικανός να τα προβλέψει. Θετικό αποτελεί το γεγονός ότι η βασική λειτουργία του τσιπ δε μπορεί να παραβιαστεί, ωστόσο, χρειάζεται προσοχή ώστε να μην υπάρξει πρόσθεση μηχανισμού ή εξαρτήματος που εμπεριέχει στοιχεία κλασικής φυσικής. Τα τσιπ παράγουν εκατομμύρια bits ανά δευτερόλεπτο, στο οποίο οφείλουν τις εξαιρετικές επιδόσεις τους στην κοινωνική δικτύωση (κλήσεις, βιντεοκλήσεις, κρυπτογράφηση μεγάλου όγκου δεδομένων) ενώ το μικρό μέγεθός τους είναι κατάλληλο για την τοποθέτησή τους στο εσωτερικό άλλων συσκευών (υπολογιστών, κινητών κ.α.). Αυτά τα τσιπ αναμένεται να εκδοθούν στην αγορά, γεγονός που θα βελτιστοποιήσει την αξιοπιστία δισεκατομμυρίων καθημερινής χρήσης συσκευών. Ως αποτέλεσμα ο τομές της ασφάλειας και της άμυνας θα λάβει σημαντικά οφέλη, και ήδη παρέχονται χρηματοδοτήσεις σε πληθώρα ερευνητικών ομάδων προς επίτευξη αυτού του σκοπού.

Σε κάθε περίπτωση ένα ουσιώδης φύσης ζήτημα που προκύπτει είναι η εκτίμηση της εντροπίας του παραγωγού τυχαιότητας, δηλαδή των ακατέργαστων bit που δημιουργούνται για την εξαγωγή των πραγματικών. Παρά το γεγονός ότι έχουν αναπτυχθεί εξελιγμένες τεχνικές εκτίμησης εντροπίας, οι μέθοδοι είναι δύσκολο να εφαρμοστούν[09,48].

4.12 Μειονεκτήματα

Παρά τη πληθώρα δυνατοτήτων που παρέχει απλόχερα η κβαντική κρυπτογραφία, υπάρχουν πλεονεκτήματα που κρίνεται απαραίτητο να ληφθούν υπ'οψη προς δημιουργία νέων βέλτιστων μοντέλων:

- Η κβαντική κρυπτογραφία είναι ακριβή στη διεξαγωγή της και με την τρέχουσα τεχνολογία που χρησιμοποιείται δε μπορεί να θεωρείται οικονομικά αποδοτική.

- Τα Qubit είναι εξαιρετικά ευαίσθητα και μπορούν να επηρεαστούν μικροσκοπικούς θορύβους και δονήσεις όπως ηλιακό φως, ήχος, κοψίματα, στροφές στο καλώδιο κλπ. Αυτά είναι ικανά να αλλάξουν το spin του πολωμένου φωτονίου (qubit) αλλάζοντας έτσι τα δεδομένα που είναι κωδικοποιημένα σε αυτό.

- Η σπατάλη πολλών qubits συμβαίνει κατά τη διάρκεια διόρθωσης κβαντικών σφαλμάτων, αφήνοντας έτσι μόνο λίγα qubits να εκτελέσουν την πραγματική υπολογιστική εργασία.

- Η κβαντική κρυπτογραφία δε μπορεί να διεξαχθεί σε μεγάλες αποστάσεις, γιατί όσο μεγαλύτερη είναι η απόσταση, τόσος μεγαλύτερος είναι ο θόρυβος που μπορεί να επηρεάσει τα qubits και να αλλάξει τη πόλωσή τους[36].

4.13 Μετα-κβαντική κρυπτογραφία

Η μετα-κβαντική κρυπτογραφία (πολλές φορές αναφέρεται και ως κβαντικά-απόδεικτική, κβαντικά-ανθεκτική, κβαντικά ασφαλής) αναφέρεται σε κρυπτογραφικούς αλγορίθμους (συνήθως αλγορίθμους δημόσιου κλειδιού) που θεωρείται ότι είναι ασφαλείς έναντι επιθέσεων που διεξάγονται από κβαντικό υπολογιστή. Δημιουργήθηκε με αφορμή το γεγονός ότι μια ενδεχόμενη ολοκλήρωση των κβαντικών υπολογιστών θα διαταράξει τις συνθήκες ασφάλειας δεδομένων και συστημάτων όπως την ακεραιότητα (αυθεντικοποίηση) δεδομένων κατά τη μετάδοσή τους, αλλά και τη συνθήκη της εμπιστευτικότητας που εξασφαλίζεται με την κρυπτογραφία[18]. Αν και η τρέχουσα δημοσιως γνωστή πειραματική κβαντική υπολογιστική

δεν είναι αρκετά ισχυρή ώστε να επιτεθεί σε πραγματικά κρυπτοσυστήματα, πολλοί κρυπτογράφοι ερευνούν νέους αλγορίθμους σε πείπτωση που ο κβαντικός υπολογισμός γίνει απειλή στο μέλλον[45]. Συνεπώς, στόχος της μετα-κβαντικής κρυπτογραφίας είναι η ανάπτυξη κρυπτογραφικών συστημάτων που είναι ασφαλή τόσο έναντι κβαντικών όσο και κλασικών υπολογιστών, και μπορούν να διαλειτουργήσουν με υπάρχοντα πρωτόκολλα και συστήματα επικοινωνιών. Σε αντίθεση με την απειλή που θέτει ο κβαντικός υπολογισμός στους τρέχοντες αλγόριθμους δημόσιου κλειδιού, οι περισσότεροι σύγχρονοι κρυπτογραφικοί αλγόριθμοι και οι λειτουργίες κατακερματισμού θεωρούνται σχετικά ασφαλείς έναντι επιθέσεων από κβαντικούς υπολογιστές. Η μετα-κβαντική κρυπτογραφία και η κβαντική κρυπτογραφία αποτελούν διαφορετικού τύπου κρυπτογραφικά συστήματα που είναι σχεδιασμένα για αντιμετώπιση διαφορετικών ζητημάτων ασφάλειας. Πιο συγκεκριμένα κάποιες από τις κεντρικές διαφορές τους:

1. Η κβαντική κρυπτογραφία βασίζεται της κβαντικής φυσικής ενώ η μετα-κβαντική κρπτογραφία βασίζεται σε κλασικούς αλγορίθμους και μαθηματικά προβλήματα[45].

2. Η κβαντική κρυπτογραφία είναι ευάλωτη σε επιθέσεις κβαντικών υπολογιστών σε αντίθεση με τη μετα-κβαντική που κατασκευάζεται με βασική αρχή την ανθεκτικότητα έναντι αυτών[46].

3. Η κβαντική κρυπτογραφία απαιτεί εξειδικευμένο υλικό όπως ανιχνευτές φωτονίων, λέιζερ και συσκευές διανομής κβαντικού κλειδιού, που τη καθιστούν δαπανηρή. Από την άλλη η μετα-κβαντική κρυπτογραφία μπορεί να εφαρμοστεί χρησιμοποιώντας ένα τυπικό λογισμικό και hardware[46].

4. Η κβαντική κρυπτογραφία παρέχει αποδεδειγμένη ασφάλεια, πράγμα που σημαίνει ότι μπορεί να αποδειχθεί μαθηματικά η ανικανότητα ενός υποκλοπέα να λάβει κάποια πληροφορία από το κρυπτογραφημένο μήνυμα. Από την άλλη η μετα-κβαντική κρυπτογραφία βασίζεται στην υπόθεση ότι ορισμένα μαθηματικά είναι δύσκολο να επιλυθούν (πρόβλημα παραγοντοποίησης ακεραίων, διακριτού λογαρίθμου κλπ.)[45]

Οι διαδικασίες εύρεσης των αλγορίθμων αυτών έχουν τραβήξει το ενδιαφέρον ακαδημαϊκών και βιομηχανιών μέσω μίας σειράς συνεδρίων PQCrypto από το 2006, και πρόσφατα από διάφορα εργαστήρια για την Κβαντική Ασφαλή Κρυπτογραφία που φιλοξενούνται από το Ευρωπαϊκό Ινστιτούτο Τηλεπικοινωνιακών Προτύπων (ETSI) και το Ινστιτούτο Κβαντικής Πληροφορικής[24]. Συνοπτικά τόσο η κβαντική όσο και η μετα-κβαντική κρυπτογραφία είναι συστήματα σχεδιασμένα ώστε να παρέχουν ασφάλεια, αλλά διαφέρουν ως προς τις υποκείμενες μαθηματικές αρχές,τους μηχανισμούς ανταλλαγής κλειδιών, τις απαιτήσεις υλοποίησης και την αντίσταση σε κβαντικούς υπολογιστές.

Επί του παρόντος η μετα-κβαντική κρυπτογραφία βασίζεται στις παρακάτω προσεγγίσεις:

1. Κρυπτογραφία βασισμένη σε πλέγμα: που περιλαμβάνει εκμάθηση με σφάλματα, εκμάθηση δακτυλίου με σφάλματα, τα παλιότερα NTRU ή GGH σχήματα κωδικοποίησης κ.α.

2. Τη πολυμεταβλητή: Που περιλαμβάνει κρυπτογραφικά συστήματα όπως το Rainbow που βασίζεται στη δυσκολία επίλυσης συστημάτων πολυμεταβλητών εξισώσεων.

3. Βασισμένη σε κατακερματισμό: Περιλαμβάνει κρυπτογραφικά συστήματα όπως οι υπογραφές Lamport, το σχήμα υπογραφών Merkle, το XMSS, το SPHINCS, και τα σχήματα WOTS.

4. Βασισμένη σε κώδικα: Τα κρυπτογραφικά συστήματα βασίζονται σε κώδικες διόρθωσης σφαλμάτων, όπως οι αλγόριθμοι κρυπτογράφησης McEliece και Niederreiter και το σχετικό σχήμα υπογραφής Courtois, Finiasz και Seider.

5. Βασισμένη στην ισογένεση: Κρυπτογραφικά συστήματα που βασίζονται στις ιδιότητες ισογονικών γραφημάτων ελλειπτικών καμπυλών (και αβελιανών ομάδων υψηλότερης διάσταση) σε πεπερασμένα πεδία.

6. Κβαντική αντίσταση συμμετρικού κλειδιού: Εφόσον κάποιος χρησιμοποιεί αρκετά μεγάλα μεγέθη κλειδιών[44].

Κεφάλαιο 5
Επίλογος-Μελλοντικές Κατευθύνσεις

5.1 Πρακτικές προκλήσεις για το μέλλον

Πρωτού γίνουν ευρέως αποδεκτά διάφορα συστήματα QKD, υπάρχει μία πληθώρα ουσιαστικών προκλήσεων που θα πρέπει να νατιμετωπιστούν σε κάθε περίπτωση. Αυτές είναι που θα πρέπει να ληφθούν υπ'όψη σοβαρά από μελλοντικούς ερευνητές που θα επιθυμούν να εφεύρουν νέα πρωτόκολλα ή να βελτιστοποιήσουν τα υπάρχοντα:

- Ρυθμός κλειδιού: Τα κλειδιά κρυπτογράφησης που δημιουργούνται από το QKD μπορούν να χρησιμοποιηθούν σε συμμετρικό σχήμα κρυπτογράφησης όπως το AES (που είναι κβαντικά ανθεκτικό) για ενισχυμένη ασφάλεια ή μπορούν να συνδυαστούν με ένα one-time-pad σχήμα κρυπτογράφησης μιας εφαρμογής για ανευ όρων ασφάλεια. Και στις δύο περιπτώσεις ο ασφαλής ρυθμός κλειδιού που επιτυγχάνεται από το υποκείμενο επίπεδο QKD σε ένα τυπικό σενάριο είναι ζωτικής σημασίας. Υψηλότεροι ρυθμοί ασφάλειας

επιτρέπουν πιο συχνή ενημέρωση του κλειδιού κρυπτογράφησης σε συμμετρικές μεθόδους.

- Απόσταση: Η επέκταση του εύρους επικοινωνίας των συστημάτων QKD είναι σημαντικός κινητήριος παράγοντας για τις τεχνολογικές εξελίξεις εν'όψει μελλοντικών εφαρμογών δικτύου. Συστήματα QKD βασισμένα σε ανίχνευση μονού φωτονίου υπερασπίζεται την απόσταση επικοινωνίας από σημείο σε σημείο (ή απώλεια καναλιού). Εδώ είναι ο χαμηλός θόρυβος των ανιχνευτών ενός φωτονίου ο βασικός παράγοντας ενεργοποίησης. Ειδικότερα εξαρτάται το εφικτό εύρος σχετικά με τον τύπο και τη θερμοκρασία των ανιχνευτών.

- Κόστος και στιβαρότητα: Για συστήματα QKD που χρησιμοποιούνται στον πραγματικό κόσμο, το χαμηλό κόστος και η στιβαρότητα αποτελούν απαραίτητα χαρακτηριστικά παράλληλα με τη παροχή υψηλών επιδόσεων[42].

5.2 Μελλοντικοί τομείς αξιοποίησης

Η κβαντική κρυπτογραφία αναμένεται να αναπτυχθεί γρήγορα λόγω των απαιτήσεων της σε διάφορους κλάδους λόγω των αυξανόμενων ανησυχιών σχετικά με τα δεδομένα και την ιδιωτικότητα. Όλο και περισσότερες μεγάλες εταιρίες παρέχουν υπηρεσίες κβαντικής κρυπτογραφίας. Αυτό ήταν αναμενόμενο καθώς τα εργαλεία της κβαντικής κρυπτογραφίας αναμένεται στο μέλλον να απαιτούνται για τη κάλυψη αναγκών ορισμένων βιομηχανιών μεταξύ των οποίων:

1) Τραπεζική και οικονομία: Οι τράπεζες οφείλουν να ενημερώνουν διαρκώς τα συστήματα ασφάλειάς τους για να συμβαδίζουν διαρκώς με τις αναπτυσσόμενες τεχνολογίες, ώστε να διασφαλίζουν τη διαθεσιμότητα γρήγορων και αξιόπιστων δεδομένων.

2) Cloud και αποθήκευση δεδομένων: Όπως είναι γνωστό πολλές εταιρίες καταφεύγουν σε αποθηκευτικούς χώρους cloud αντί να έχουν τα δικά τους κέντρα δεδομένων. Οι

απαιτήσεις για γρήγορα και ασφαλή μέσα μετάδοσης έχει γίνει απαραίτητη για τη διασφάλιση προστασίας των ευαίσθητων δεδομένων κάθε εταιρίας.

3) Κυβέρνηση και άμυνα: Με την ανάπτυξη κβαντικών υπολογιστών, εμπιστευτικά δεδομένα μιας χώρας θα θεωρούνται όλο και λιγότερο ασφαλή και ενδέχεται να υπόκεινται σε κυβερνο-επιθέσεις. Απαιτείται κβαντική κρυπτογραφία για την ασφάλεια αυτών των δεδομένων.

4) Υγειονομική περίθαλψη: Τα δεδομένα υγειονομικής περίθαλψης περιέχουν ζωτικής σημασίας πληροφορίες σχετικά με έναν ασθενή, επομένως πρέπει να διασφαλιστεί ότι τα δεδομένα μένον ασφαλή ακόμα και κατά τη μετάδοση μέσω ενός δικτύου.

5) Τηλεπικοινωνίες: Η τηλεπικοινωνία αποτελεί τη βάση των μέσων επικοινωνίας και μετάδοσης ζωτικής σημασίας πληροφοριών, όπως άρθρα ειδήσεων που αναμένεται να δημοσιευτούν ή ειδήσεις για το χρηματιστήριο που λαμβάνουν χώρα στα τηλεπικοινωνιακά δίκτυα κάθε μέρα. Έτσι, η απαίτηση ασφαλών και γρήγορων μέσων τηλεπικοινωνιών, απαιτούν τη δημιουργία κβαντικών μέσων τηλεπικοινωνιών[36].

5.3 Επίλογος

Η πρόοδος στον τομέα των μαθηματικών και η ανάπτυξη μεθόδων κβαντικού υπολογισμού έχουν οδηγήσει σε μείωση της ασφάλειας των κλασικών κρυπτογραφικών συστημάτων. Σε αυτά τα πλαίσια η κβαντική κρυπτογραφία παρουσιάζεται ως η απόλυτα ασφαλής κρυπτογραφική μέθοδος, λόγω της ικανότητας ανίχνευσης υποκλοπής καθώς βασίζεται στη φυσική και όχι σε μαθηματικές μεθόδους. Στη παρούσα διατριβή έγινε εκτενής ανάλυση αυτής, με τη θεωρητική δόμηση των κυριότερων πρωτοκόλλων της (BB84 και B92) και την υλοποίησή τους σε προγραμματιστικό περιβάλλον προς εξαγωγή χρήσιμων συμπερασμάτων. Με τα χρήσιμα ευρήματα που έδωσε στην επιστημονική κοινότητα όλο και αυξάνεται από αυτήν η άποψη ότι η κβαντική κρυπτογραφία θα είναι το τελικό σημείο εξέλιξης της κρυπτογραφίας. Ωστόσο, σε αυτήν την ιδιαίτερα απαιτητική πορεία που ακολούθησα προς την υλοποίηση αυτής της διατριβής, με όσα βιβλιογραφικά εργαλεία συνέλεξα για τη δημιουργία της, τάσσομαι υπέρ διαφορετικής άποψης. Αρχικά, η κβαντική κρυπτογραφία δεν ήρθε για να αντικαταστήσει τις κλασικές μεθόδους κρυπτογραφίας, αλλα για να σταθεί συμπληρωματικά δίπλα σε αυτές. Επιπρόσθετα, για την επιβεβαίωση της ασφάλειας ενός QKD συστήματος, είναι απαραίτητη η επαλήθευση των υποθέσεων που γίνονται στις αποδείξεις ασφάλειας και σε πρακτικό σύστημα. Ακόμα και μεταξύ των QKD

πρωτοκόλλων υπάρχουν πολλοί παράγοντες που μπορούν να επηρεάσουν την απόδοση τους όπως φάνηκε και στην πρακτική εφαρμογή που πραγματοποιήθηκε παραπάνω, οι οποίοι θα πρέπει να συνεκτιμώνται σε κάθε περίπτωση. Παράλληλα με την βελτίωση των μεθόδων QKD- όπως αναμένεται- ένας επίδοξος λαθρακουστής θα μπορούσε να επενδύσει αρκετά στη κβαντική τεχνολογία ώστε να εκμεταλλευτεί κενά που μπορεί να υπάρχουν σε ένα πρακτικό σύστημα QKD. Ζούμε σε μία συναρπαστική εποχή όπου η αλληλεπίδραση μεταξύ θεωρίας και πράξης της κβαντικής κρυπτογραφίας μόλις ξεκίνησε, και ο αιώνιος πόλεμος μεταξύ όσων κατασκευάζουν κώδικα και όσων τον «σπάνε» θα συνεχίζεται...

Βιβλιογραφία

[01] Πουλάκης, Δ. (2015). *Υπολογιστική θεωρία αριθμών* [Προπτυχιακό εγχειρίδιο]. Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις.

[02] Κοντογεώργης, Α., & Αντωνιάδης, Ι. (2015). *Πεπερασμένα σώματα και κρυπτογραφία* [Προπτυχιακό εγχειρίδιο]. Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις.

[03] Δανάη-Μαρία Νείλα, Κρυπτογραφία και Ασφάλεια συστημάτων, Διπλωματική εργασία, 2021

[04] Παγουρτζής, Α., & Ζάχος, Ε. (2015). *Υπολογιστική κρυπτογραφία* [Προπτυχιακό εγχειρίδιο]. Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις.

[05] Σημειώσεις Θεωρίας Αριθμών, Μιχάλης Παπαδημητράκης,Τμήμα Μαθηματικών,Πανεπιστήμιο Κρήτης

[06] Χριστίνα Ζερβοπούλου, Κρυπτογραφια βασισμενη σε δικτυωματα ακεραιων: θεωρια και εφαρμογες, Διπλωματικη εργασία, 2013

[07] Κρυπτογραφία-Βικιπαίδεια

[08] Cryptography-Wikipedia

[09] Γεωργία-Μαρία Καρανίκα, Η Κβαντική Τεχνολογία Στην Αμυντική Έρευνα, Διπλωματικη εργασία, 2019

[10] Προυσάλης Κωνσταντίνος, Κβαντική Κρυπτογραφία & Κβαντική Κρυπτανάλυση, Διπλωματική εργασία, 2008

[11] Ανάπτυξη εφαρμογής ασφαλούς επικοινωνίας Δόλλας Νικόλαος , Διπλωματική εργασία, 2017

[12] Δημήτριος Τσακτσήρας, Κρυπτογραφία και Ελλειπτικές Καμπύλες, Διπλωματικη εργασία, 2011

[13] Δημήτριος Πίτος, Κβαντική κρυπτογράφηση, Διπλωματική εργασία, 2017

[14] Άννα Ελένη Γεωργοπούλου, Μελέτη Πρωτοκόλλων Κρυπτογραφίας, Διπλωματική εργασία, 2015

[15] Μαυρίδης, Ι. (2015). *Ασφάλεια πληροφοριών στο διαδίκτυο* [Προπτυχιακό εγχειρίδιο]. Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις.

[16] DES-Βικιπαίδεια

[17] AES-Wikipedia

[18] Βουδούρης Αναστάσιος, Κβαντικη και Μετα-Κβαντικη Κρυπτογραφια και Εφαρμογές, Διπλωματική εργασία, 2019

[19] Κβαντικη Μηχανικη-Βικιπαιδεια

[20] Γιώργος Ασημάκης, Κβαντικοι Υπολογιστες, Διπλωματικη εργασία, 2015

[21] Παναγιώτης Γρηγοριάδης, Κβαντικοι Υπολογιστες και Κβαντικη Υπολογισιμοτητα, Διπλωματική εργασία, 2020

[22] Γιώργος Κάππος, Κβαντικοί Υπολογιστές,Διπλωματική εργασία, 2016

[23] Bloch sphere-Wikipedia

[24] Μαρία Χριστίνα Δριτσοπούλου, Μετα-κβαντικη κρυπτογραφία, Διπλωματική εργασία, 2019

[25] Lo, Hoi-Kwong, and Yi Zhao. "Quantum cryptography." *arXiv preprint arXiv:0803.2507* (2008).

[26] Μαρία Τσιπουρίδη, Κβαντική κρυπτογραφία - Hardware

[27] Zbinden, H., et al. "Quantum cryptography." *Applied Physics B: Lasers & Optics* 67.6 (1998).

[28] Bruss, Dagmar, et al. "Quantum cryptography: A survey." *ACM Computing Surveys (CSUR)* 39.2 (2007): 6-es.

[29] Quantum key distribution-Wikipedia

[30] https://github.com/Lee-sun-ah/Quantum_Cryptography/tree/master/BB84%20protocol

[31] https://github.com/Lee-sun-ah/Quantum_Cryptography/tree/master/B92%20protocol

[32] Mafu, Mhlambululi, and Makhamisa Senekane. "Implementation and Security Analysis of the B92 Protocol using Id3100 Clavis2 System." *Applied Mathematics & Information Sciences* 15.5 (2021): 661-666.

[33] Elboukhari, Mohamed, Mostafa Azizi, and Abdelmalek Azizi. "Quantum Key Distribution Protocols: A Survey." *International Journal of Universal Computer Science* 1.2 (2010).

[34] Gopal, Ayush. "Experiments with B92 Quantum Key Distribution Algorithm Implementation." (2022).

[35] Pirandola, Stefano, et al. "Advances in quantum cryptography." *Advances in optics and photonics* 12.4 (2020): 1012-1236.

[36] Pranav Mendiratta, "Quantum Cryptography." IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 8 Issue 1, January 2021

[37] BBM92-Wikipedia

[38] Six-state protocol-Wikipedia

[39] SARG04-Wikipedia

[40] Singh, Hitesh, Dharmendra Lal Gupta, and Ashish Kumar Singh. "Quantum key distribution protocols: a review." *Journal of Computer Engineering* 16.2 (2014): 1-9.

[41] Gheorghieș, Alexandru-Ștefan, Darius-Marian Lăzăroi, and Emil Simion. "A Comparative Study of Cryptographic Key Distribution Protocols." *Cryptology ePrint Archive* (2021).

[42] Diamanti, Eleni, et al. "Practical challenges in quantum key distribution." *npj Quantum Information* 2.1 (2016): 1-12.

[43] Diffie-Hellman-Wikipedia

[44] Post-quantum cryptography,Wikipedia

[45] Tan, Xiaoqing. "Introduction to quantum cryptography." *Theory and Practice of Cryptography and Network Security Protocols and Technologies* (2013).

[46] Chen, Lily, et al. *Report on post-quantum cryptography*. Vol. 12. Gaithersburg, MD, USA: US Department of Commerce, National Institute of Standards and Technology, 2016.

[47] Lo, Hoi-Kwong, and Norbert Lütkenhaus. "Quantum cryptography: from theory to practice." *arXiv preprint quant-ph/0702202* (2007).

[48] Shenoy-Hejamadi, Akshata, Anirban Pathak, and Srikanth Radhakrishna. "Quantum cryptography: Key distribution and beyond." *Quanta* 6.1 (2017): 1-47.

[49] https://github.com/cotaylor/qkdsim/tree/master/qkdsim