Αξιοποίηση Κρυπτογραφικών Τεχνικών (Υπερ)Ελλειπτικών Καμπυλών για Βελτίωση της Ιδιωτικότητας σε Αυτό-Οργανούμενα Δίκτυα Οχημάτων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Παναγιώτης Ντάγκας

Επιβλέπουσα: Ιωάννα Ρουσσάκη
Αναπλ. Καθηγήτρια Ε.Μ.Π.

Αθήνα 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Αξιοποίηση Κρυπτογραφικών Τεχνικών (Υπερ)Ελλειπτικών Καμπυλών για Βελτίωση της Ιδιωτικότητας σε Αυτό-Οργανούμενα Δίκτυα Οχημάτων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Παναγιώτης Ντάγκας

Επιβλέπουσα: Ιωάννα Ρουσσάκη
Αναπλ. Καθηγήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 7η Ιουλίου 2023.

.....	.....	.....
Ιωάννα Ρουσσάκη	Μιλτιάδης Αναγνώστου	Συμεών Παπαβασιλείου
Αναπλ. Καθηγήτρια Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα 2023

.....
Παναγιώτης Ντάγκας
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π

Περίληψη

Το Διαδίκτυο των Αντικειμένων, ή αλλιώς Internet of Things, γνωρίζει ραγδαία εξέλιξη στην έρευνα, αλλά και στην βιομηχανία τα τελευταία χρόνια, δίνοντας την ευκαιρία για την ανάπτυξη έξυπνων και διαλειτουργικών δικτύων συσκευών. Πιο συγκεκριμένα, έδωσε τη δυνατότητα για την δημιουργία και εξέλιξη των Αυτό-Οργανούμενων Δικτύων για Οχήματα (Vehicular Ad Hoc Networks – VANETs), τα οποία αξιοποιούνται κυρίως για τη διασφάλιση της ασφάλειας των οδηγών και την μείωση των τροχαίων ατυχημάτων. Ωστόσο, αυτή η ραγδαία εξέλιξη διεγείρει σημαντικές ανησυχίες για την διασφάλιση της ιδιωτικότητας των χρηστών, μιας και παρατηρείται αναλογικά και η αύξηση των κακόβουλων ατόμων ή οργανισμών για την κλοπή των δεδομένων, πράγμα που μπορεί να εγείρει σοβαρούς κινδύνους στους οδηγούς κατά την μετακίνησή τους στις νέες Έξυπνες Πόλεις.

Στην παρούσα εργασία διερευνώνται και αξιολογούνται τα σύγχρονα σχήματα προστασίας της ιδιωτικότητας των χρηστών των VANETs, με βάση την ασφάλεια αλλά και την αποδοτικότητά τους, αφού τα περιβάλλοντα των VANETs παρέχουν περιορισμένους πόρους προς αξιοποίηση. Με σκοπό την επιτάχυνση και την μείωση των απαιτούμενων μηνυμάτων για τη διασφάλιση της ιδιωτικότητας, αρχικά γίνεται μία παρουσίαση της σύγχρονης βιβλιογραφίας για τις πιο δημοφιλής τεχνικές, στη συνέχεια διερευνάται η χρήση των υπερελλειπτικών καμπυλών για κρυπτογραφικούς σκοπούς σε περιβάλλοντα περιορισμένων πόρων και τέλος πραγματοποιούνται προσομοιώσεις 3 κρυπτογραφικών αλγορίθμων (ECC, HECC genus 2, HECC genus 3) πάνω σε ένα σχήμα αποδοτικής αυθεντικοποίησης και ασφαλής μετάδοσης μηνυμάτων σε VANETs, με σκοπό την εξαγωγή συμπερασμάτων για την αξιοποίηση του κάθε αλγόριθμου στο συγκεκριμένο πεδίο εφαρμογών.

Λέξεις-Κλειδιά

Διαδίκτυο των Αντικειμένων (IoT), Διαδίκτυο των Οχημάτων (IoV), Αυτό-Οργανούμενα Δίκτυα για Οχήματα (VANETs), ιδιωτικότητα οχημάτων, ψηφιακά πιστοποιητικά, ψηφιακή υπογραφή, ECC, HECC, κατανάλωση ενέργειας, μέγεθος μηνύματος

Abstract

The Internet of Things, or IoT, has been experiencing rapid evolution in both research and industry in recent years, offering the opportunity for the development of smart and interoperable device networks. More specifically, it has enabled the creation and advancement of Vehicular Ad Hoc Networks (VANETs), which are primarily utilized to ensure driver safety and reduce traffic accidents. However, this rapid evolution raises significant concerns regarding the privacy of users, as there is a proportional increase in malicious individuals or organizations seeking to steal data, which can pose serious risks to drivers during their movement through Smart Cities.

This study explores and evaluates contemporary privacy protection schemes for VANET users, based on their security and efficiency, considering that VANET environments provide limited resources for utilization. In order to speed up and reduce the required messages for privacy preservation, an overview of the modern literature on the most popular techniques is presented, followed by an investigation into the use of elliptic curves for cryptographic purposes in resource-constrained environments. Lastly, simulations of three cryptographic algorithms (ECC, HECC genus 2, HECC genus 3) are performed on an efficient authentication and secure message transmission scheme in VANETs, with the aim of drawing conclusions about the utilization of each algorithm in this specific field of application.

Keywords

Internet of Things (IoT), Internet of Vehicles (IoV), Vehicular Ad-hoc Networks (VANETs), vehicle privacy, digital certificates, digital signatures, ECC, HECC, energy consumption, message size

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά την επιβλέπουσα καθηγήτρια κα. Ιωάννα Ρουσσάκη για την ευκαιρία που μου έδωσε να ασχοληθώ με το πολύ ενδιαφέρον αντικείμενο της παρούσας εργασίας, καθώς και για την πολύτιμη βοήθεια της και την καθοδήγησή της κατά την εκπόνηση της. Επιπλέον, θα ήθελα να ευχαριστήσω τον διδακτορικό ερευνητή κ. Γεώργιο Ρούτη για την καθοδήγηση και τη βοήθειά του κατά τη διάρκεια της υλοποίησης της εργασίας.

Επίσης, επιθυμώ να εκφράσω τις ευχαριστίες μου στην οικογένειά μου για την αγάπη και τη στήριξη που έδειξαν σε όλα τα χρόνια των σπουδών μου και μου έδιναν πάντοτε κίνητρο για να συνεχίζω.

Τέλος, ευχαριστώ τους φίλους μου που μου στάθηκαν σε όλες τις δυσκολίες που αντιμετώπισα, προσφέροντάς μου αναντικατάστατη δύναμη και αγάπη. Ιδιαίτερη μνεία οφείλω στην Ράνια για την πολύτιμη βοήθεια της τα τελευταία χρόνια και στην κοπέλα μου, Εστέλ, για την υπομονή και τη στήριξή της.

Πίνακας Περιεχομένων

Πίνακας Εικόνων

Περιεχόμενα Πινάκων

Πίνακας ακρωνυμίων

Μερικές από τις βασικές έννοιες που θα συναντηθούν αρκετά συχνά στο υπόλοιπο της εργασίας:

Πίνακας 1: Πίνακας Ακρωνυμίων

Ακρωνύμιο	Περιγραφή
AES	Advanced Encryption Scheme (Αλγόριθμος συμμετρικής κρυπτογράφησης)
AODV	Ad hoc On-Demand Distance Vector Routing (Πρωτόκολλο δρομολόγησης πακέτων σε VANETs)
ASCII	American Standard Code for information Interchange
CA	Certificate Authority (Αρχή έκδοσης πιστοποιητικών)
CBC	Cipher Block Chaining
CH	Cluster Head (Κεφαλή ομάδας)
CRL	Certificate Revocation List (Λίστα Ανάκλησης Πιστοποιητικών)
CSMA/CA	Carrier-sense multiple access with collision avoidance
DDoS	Distributed Denial of Service (Κατανεμημένη Άρνηση Υπηρεσίας)
DES	Data Encryption Standard
3DES	Triple Data Encryption Standard
DID	Dummy ID
DLP	Discrete Logarithm Problem (Πρόβλημα Διακριτού Λογάριθμου)
DMV	Department of Motor Vehicles (Υπουργείο Μεταφορών)
DoS	Denial of Service (Άρνηση Υπηρεσίας)
ECC	Elliptic Curve Cryptography (Κρυπτογραφία ελλειπτικών καμπυλών)
ECIES	Elliptic Curve Integrated Encryption Scheme (Αλγόριθμος κρυπτογράφησης βασισμένη σε ανταλλαγή Diffie-Hellman και συμμετρικό κλειδί)

FCS	Frame Check Sequence
FTP	File Transfer Protocol
GL	Group Leader (βλ. σχήμα [28]) (Ο αρχηγός ομάδας)
GF	Galois Field (Σώμα Galois)
HECC	Hyperelliptic Curve Cryptography (Κρυπτογραφία υπερελλειπτικών καμπυλών)
(H)ECDH	(Hyper) Elliptic Curve Diffie-Hellman (Ανταλλαγή κλειδιού Diffie-Hellman βασισμένη στην κρυπτογραφία (υπερ)ελλειπτικών καμπυλών)
(H)ECDLP	(Hyper) Elliptic Curve Discrete Logarithm Problem (Κρυπτογραφία (υπερ)ελλειπτικών καμπυλών βασισμένη στον Διακριτό Λογάριθμο)
(H)ECDSA	(Hyper) Elliptic Curve Digital Signature Algorithm (Ψηφιακές υπογραφές (υπερ)ελλειπτικών καμπυλών DSA)
(H)ECQV	(Hyper) Elliptic Curve Qu-Vanstone (Ψηφιακά πιστοποιητικά (υπερ)ελλειπτικών καμπυλών Qu-Vanstone)
HKDF	Hash-based Key-Derivation Function (Συνάρτηση παραγωγής κλειδιών βασισμένη στον κατακερματισμό)
HTTPS	Hypertext Transfer Protocol Secure
ID	Ταυτότητα ή ψευδώνυμο οχήματος / RSU
IP	Internet Protocol
IPSec	Internet Protocol Secure
ITS	Intelligent Transportation Systems (Έξυπνα Συστήματα Μεταφοράς)
IoT	Internet of Things (Διαδίκτυο των Αντικειμένων)
IoV	Internet of Vehicles (Διαδίκτυο των Οχημάτων)
IV	Initialization Vector (AES)
MAC (network)	Media Access Control (Έλεγχος Προσπέλασης στο Μέσο)
MAC (security)	Message Authentication Code (Κώδικας Αυθεντικοποίησης Μηνύματος)

ML	Montgomery's Ladder
MLME	MAC layer management entity
NAF	Non-adjacent Form
OBU	On-Board Unit (Συσκευή που επιτρέπει την ασύρματη επικοινωνία ενός οχήματος)
OFDM	Orthogonal frequency-division multiplexing (Ορθογωνική Πολυπλεξία Διαίρεσης Συχνότητας)
PKCS	Public Key Cryptography Standards
QoS	Quality of Service (Ποιότητα Υπηρεσίας)
RC	Reginal Cloud (Τοπικό Σύννεφο)
RSA	Rivest-Shamir-Adleman (Αλγόριθμος ασύμμετρης κρυπτογράφησης)
RSU	Roadside Unit (Μονάδα υποδομής στο δίκτυο VANET)
Rx	Receive (Λήψη)
SAM	Square and Multiply
SHA	Secure Hash Algorithm
SSH	Secure Shell Protocol
TA	Trusted Authority (Η Έμπιστη Αρχή, συνήθως ταυτίζεται με την CA)
TCP	Transmission Control Protocol (Πρωτόκολλο Ελέγχου Μεταφοράς)
TLS	Transport Layer Security
TPD	Tamper-Proof Device (Αμετάβλητη συσκευή)
Tx	Transmit (Μετάδοση)
UDP	User Datagram Protocol
VANET	Vehicular ad hoc Network (Αυτό-Οργανούμενο Δίκτυο Οχημάτων)
V2V/V2I	Vehicle 2 Vehicle/Infrastructure (Επικοινωνία Οχήματος με Όχημα/Υποδομή)
WAVE	Wireless Access in Vehicular Environments
WSMP	WAVE Short Message Protocol

Κεφάλαιο 1: Εισαγωγή

Το παρόν κεφάλαιο εστιάζει στο κίνητρο που οδήγησε στην εκπόνηση της διπλωματικής εργασίας, στους στόχους που προσπαθεί να επιτύχει και στον τρόπο που οργανώνεται το κείμενό της.

1.1 Κίνητρο έρευνας

Το Διαδίκτυο των Αντικειμένων, ή αλλιώς IoT, αναφέρεται σε ένα δίκτυο από φυσικές συσκευές, οχήματα, εφαρμογές ή και άλλα αντικείμενα που είναι ενσωματωμένα με αισθητήρες ή λογισμικό, που αποστέλλουν τα δεδομένα τους στο διαδίκτυο. Μία υποκατηγορία του Διαδικτύου των Αντικειμένων είναι το Διαδίκτυο των Οχημάτων, ή αλλιώς IoV, το οποίο δίνει τη δυνατότητα σε αυτοκίνητα, αλλά και στην υποδομή των δρόμων, να βρίσκονται σε σύνδεση με το διαδίκτυο και να μοιράζουν τα δεδομένα τους σε διάφορους δέκτες. Ιδιαίτερα δημοφιλής τα τελευταία χρόνια είναι η τεχνολογία της επικοινωνίας μεταξύ αυτοκινήτων (V2V) και επικοινωνίας μεταξύ αυτοκινήτων και υποδομής (V2I). Με τη ραγδαία εξέλιξη που χαρακτηρίζει το Διαδίκτυο των Αντικειμένων και τις δυνατότητες που παρέχει στη βελτίωση της εμπειρίας των χρηστών στην καθημερινότητά τους, η έρευνα στο Διαδίκτυο των Οχημάτων εξελίσσεται αναλογικά. Οι νέες τεχνολογίες υπόσχονται αυτοματοποίηση, αποδοτικότητα και άνεση στον κλάδο των μεταφορών, αλλά το σημαντικότερο όλων την εξασφάλιση της ασφάλειας των οδηγών και την αποφυγή τροχαίων ατυχημάτων στους δρόμους, που αποτελούν ένα μεγάλο ποσοστό των θανάτων νέων ανθρώπων τη σήμερον ημέρα. Παράλληλα, η εξέλιξη των τεχνολογιών 5G και 6G που προσφέρουν πολύ μεγάλες ταχύτητες διασύνδεσης έρχονται να δημιουργήσουν μία νέα επανάσταση στον κλάδου του Διαδικτύου των Οχημάτων.

Παρόλα αυτά, αυτή η ραγδαία εξέλιξη δημιουργεί και νέους κινδύνους στην ιδιωτικότητα των χρηστών μιας και η κακόβουλοι χρήστες ανακαλύπτουν νέους τρόπους να παραβιάσουν τα προσωπικά δεδομένα των χρηστών και να τα εκμεταλλευτούν. Ιδιαίτερα στον κλάδο της επικοινωνίας των αυτοκινήτων (V2V/V2I) η ασφάλεια πρέπει να λαμβάνεται υπ’ όψη με μεγάλη προσοχή, καθώς οι επιτιθέμενοι, εκτός από τα προσωπικά δεδομένα, μπορεί να θέσουν τη ζωή ενός ή πολλαπλών οδηγών σε κίνδυνο. Επομένως, τα τελευταία χρόνια υπάρχει ιδιαίτερο ενδιαφέρον για την ανάπτυξη ενός ασφαλέστερου περιβάλλοντος επικοινωνίας, το οποίο όμως πρέπει να προσαρμοστεί στις περιορισμένες δυνατότητες που παρέχουν οι εμπλεκόμενες συσκευές κατά την επικοινωνία. Γι’ αυτό οι λύσεις της ασφάλειας στα συγκεκριμένα περιβάλλοντα και γενικότερα στο Διαδίκτυο των Αντικειμένων είναι αναγκαίο να είναι αποδοτικές, γρήγορες και να μην κατακλύζουν το δίκτυο με αχρείαστη κίνηση.

1.2 Συνεισφορά διπλωματικής

Αρχικά, στην παρούσα διπλωματική εργασία δίνεται έμφαση στην ενσωμάτωση ενός σχήματος ιδιωτικότητας σε Δίκτυα Αυτοκινήτων (VANETs) σε ένα περιβάλλον προσομοίωσης, με σκοπό την αναλυτικότερη μελέτη και κατανόηση τους, αλλά και την εξόρυξη χρήσιμων αποτελεσμάτων για την αξιολόγηση του εν λόγω σχήματος.

Κατά δεύτερον, η εργασία αποσκοπεί στην ανάλυση και υλοποίηση των κρυπτογραφικών τεχνικών των υπερελλειπτικών καμπυλών, ώστε να μπορέσουν να αξιοποιηθούν σε Δίκτυα Αυτοκινήτων και να μελετηθεί η εφαρμογή τους από την σκοπιά της αποδοτικότητας, της ταχύτητας και της δικτυακής κίνησης που δημιουργούν.

Τρίτον, η εργασία στοχεύει στην επέκταση του σχήματος ασφαλείας, ώστε να αφομοιώσει τις κρυπτογραφικές τεχνικές των υπερελλειπτικών καμπυλών με σκοπό να αξιολογηθούν σε ένα εφαρμόσιμο ασφαλές σχήμα και παράλληλα να βελτιωθούν οι υπερπαράμετροι του επιλεγμένου σχήματος. Λαμβάνονται ανάλογες μετρήσεις στο περιβάλλον της προσομοίωσης για την εξαγωγή συμπερασμάτων για τη χρήση των διαφορετικών οικογενειών καμπυλών.

1.3 Οργάνωση κειμένου

Η οργάνωση του κειμένου της εργασίας γίνεται ως εξής: Στο κεφάλαιο 2 παρουσιάζονται συνοπτικά το θεωρητικό υπόβαθρο που απαιτείται για την καλύτερη κατανόηση της εργασίας. Έμφαση δίνεται σε έννοιες κρυπτογραφίας, κρυπτογραφικά στοιχεία ελλειπτικών/υπερελλειπτικών καμπυλών, έννοιες δικτύου και μέθοδοι επιθέσεων. Στο 3^{ο} κεφάλαιο γίνεται επισκόπηση της συναφούς βιβλιογραφίας στα σχήματα ασφάλειας των VANETs. Στο 4^{ο} κεφάλαιο αναλύεται το πρόβλημα και η μοντελοποίησή του. Στο 5^{ο} κεφάλαιο αναφέρονται τα εργαλεία και οι αλγόριθμοι κρυπτογράφησης και στη συνέχεια, στο παρουσιάζεται η υλοποίηση τους μοντέλου με βάση τις κρυπτογραφικές τεχνικές των ελλειπτικών/υπερελλειπτικών καμπυλών και τέλος, στο 6^{ο} κεφάλαιο παρατίθενται και σχολιάζονται τα πειραματικά αποτελέσματα των προσομοιώσεων που πραγματοποιήθηκαν.

Κεφάλαιο 2: Συναφές Θεωρητικό Υπόβαθρο

Στο παρόν κεφάλαιο γίνεται μία σύντομη παρουσίαση του αναγκαίου θεωρητικού υπόβαθρου για την καλύτερη κατανόηση της έρευνας που πραγματοποιήθηκε στην εργασία. Αρχικά, αναλύονται οι κρυπτογραφικές τεχνικές που χρησιμοποιήθηκαν με ειδική μνεία στην κρυπτογραφία με βάση τις υπερελλειπτικές καμπύλες, στη συνέχεια παρουσιάζονται οι δικτυακές έννοιες με μεγαλύτερη έμφαση στο πρωτόκολλο WAVE. Επίσης, παρουσιάζονται οι διάφοροι τρόποι επίθεσης στα σχήματα ιδιωτικότητας. Τέλος, παρατίθεται σύντομα η ορολογία που χρησιμοποιείται στην εργασία.

2.1 Έννοιες κρυπτογραφίας

2.1.1 Συμμετρική κρυπτογράφηση

Η συμμετρική κρυπτογράφηση είναι μία από τις πιο διαδεδομένες κρυπτογραφικές τεχνικές, καθώς παρέχει μεγάλη ασφάλεια με σχετικά μικρό υπολογιστικό κόστος. Στη συμμετρική κρυπτογράφηση, οι δύο ή και περισσότερες πλευρές που επιθυμούν να επικοινωνήσουν χρησιμοποιούν ένα κοινό κλειδί, που ονομάζεται συμμετρικό ή μυστικό κλειδί. Στόχος είναι το αρχικό κείμενο, ή αλλιώς plain text, που κάποιος επιθυμεί να στείλει σε κάποιον/κάποιους άλλους χρήστες να μετατραπεί σε μία άλλη, μη κατανοητή μορφή για τον άνθρωπο, με τον μόνο τρόπο να αντιστραφεί η διαδικασία να είναι με τη γνώση του συμμετρικού κλειδιού.

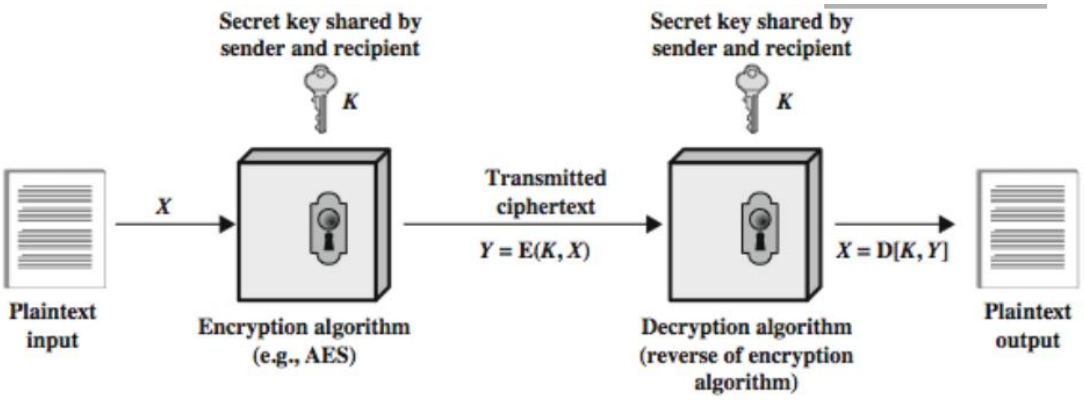
Για να επικοινωνήσει λοιπόν η Alice με τον Bob, αρχικά, πρέπει να ανταλλάξουν ένα συμμετρικό ή μυστικό κλειδί. Υπάρχουν πολλοί τρόποι να επιτευχθεί η ανταλλαγή. Ένας από αυτούς είναι να γίνει η ανταλλαγή μέσω ενός ασφαλούς καναλιού, ή μίας τεχνικής ανταλλαγής κλειδιού, όπως η τεχνική Diffie Hellman που θα παρουσιαστεί στο κεφάλαιο της ασύμμετρης κρυπτογράφησης, ή με φυσικό τρόπο (για παράδειγμα το συμμετρικό κλειδί να έχει καθοριστεί πριν τη δημιουργία του μέσου επικοινωνίας). Στη συνέχεια, οι πλευρές που επικοινωνούν πρέπει να συμφωνήσουν σε έναν αλγόριθμο κρυπτογράφησης. Οι πιο διαδεδομένοι αλγόριθμοι κρυπτογράφησης είναι:

- Advanced Encryption Standard (AES): Ο AES είναι ο πιο διαδεδομένος αλγόριθμος συμμετρικής κρυπτογράφησης, καθώς συνδυάζει υψηλή ασφάλεια και αποδοτικότητα. Ο AES είναι αλγόριθμος τμημάτων (block cipher) και κρυπτογραφεί ένα μήνυμα σε τμήματα μεγέθους 128 bit και χρησιμοποιεί κλειδιά μεγέθους 128, 192 ή 256 bit.
- Data Encryption Standard (DES): Ο DES είναι ένας σχετικά παλιός αλγόριθμος συμμετρικής κρυπτογράφησης και πλέον δεν χρησιμοποιείται, καθώς δεν

είναι αρκετά ασφαλής, μιας και χρησιμοποιεί τμήματα 64-bit και κλειδιά μεγέθους 56-bit.

- Triple Data Encryption Standard (3DES): Ο 3DES είναι μία ειδική μορφή του DES που προαναφέρθηκε και πρακτικά εφαρμόζει τον αλγόριθμος DES 3 φορές σε κάθε τμήμα, χρησιμοποιώντας 2 ή 3 κλειδιά των 56-bit. Με αυτόν τον τρόπο βελτιώνει την ασφάλεια του DES, ωστόσο δεν είναι όσο αποδοτικός όσο ο AES [1].

Αφού η Alice και ο Bob συμφωνήσουν σε αλγόριθμο κρυπτογράφησης, τότε η Alice επιχειρεί να στείλει ένα μήνυμα στον Bob πάνω σε ένα μη ασφαλές κανάλι, στο οποίο η Trudy, η οποία είναι μία κακόβουλη χρήστης, έχει πρόσβαση. Ένα τέτοιο κανάλι μπορεί να είναι δικτυακές συνδέσεις όπως WiFi, Ethernet κ.α. Η Alice, λοιπόν, κρυπτογραφεί το μήνυμα με τον επιλεγμένο αλγόριθμο κρυπτογράφησης, χρησιμοποιώντας το συμφωνηθέν κλειδί (cipher text) και το στέλνει. Ο Bob, που το λαμβάνει, για να μπορέσει να επανακτήσει το αρχικό κείμενο (plain text) χρησιμοποιεί το συμμετρικό κλειδί και τον αλγόριθμο αποκρυπτογράφησης. Αν η ανταλλαγή έγινε σωστά, τότε η ανάκτηση του μηνύματος ολοκληρώνεται με επιτυχία. Από την άλλη, η Trudy που δεν γνωρίζει το συμμετρικό κλειδί, αλλά μονάχα τον αλγόριθμο κρυπτογράφησης/αποκρυπτογράφησης δεν μπορεί να αποκτήσει κάποια πληροφορία για το μήνυμα που έχει στη διάθεσή της.



Εικόνα 1: Συμμετρική κρυπτογράφηση (Από: μάθημα Ασφάλεια Δικτύων Υπολογιστών 8ου εξαμήνου ΗΜΜΥ ΕΜΠ, Διαφάνειες Συμμετρικής Κρυπτογραφίας σελ. 9)

Αν η Trudy καταφέρει, ωστόσο, να ανακτήσει το συμμετρικό κλειδί, τότε θα έχει τη δυνατότητα να ανακτήσει την πληροφορία του αρχικού κειμένου (plain text). Επομένως, η επιλογή του αλγορίθμου, αλλά και του μεγέθους του κλειδιού παίζουν σημαντικό ρόλο στην ασφάλεια του σχήματος συμμετρικής κρυπτογράφησης. Είναι σημαντικό, αρχικά, το κλειδί να είναι αδύνατο να ανακτηθεί με τυφλή αναζήτηση (brute force attack) και γι' αυτό τον λόγο χρησιμοποιούνται κλειδιά μεγάλου μήκους. Ωστόσο, υπάρχουν επιθέσεις που βασίζονται σε αδυναμίες του αλγορίθμου, όπως οι επιθέσεις πλευρικού καναλιού (side channel attacks) που αξιοποιούν διαρροή πληροφορίας του αλγόριθμου.

Τελικά, είναι ύψιστης σημασίας οι πλευρές που επικοινωνούν να επιλέγουν συμμετρικούς αλγόριθμους που είναι αποδεδειγμένα ασφαλείς και από την άλλη να είναι αρκετά αποδοτικοί για να μπορούν να ανταπεξέλθουν στις αυξανόμενες απαιτήσεις ταχύτητας στη βιομηχανία σήμερα. Επίσης, είναι άξιο αναφοράς πως οι συμμετρικοί αλγόριθμοι κρυπτογράφησης θεωρούνται σχετικά ασφαλής απέναντι σε επιθέσεις κβαντικού υπολογισμού (post quantum secure) σε αντίθεση με αυτούς της ασύμμετρης κρυπτογράφησης [38].

2.1.2 Ασύμμετρη κρυπτογράφηση

Αντίθετα από τους αλγόριθμους κρυπτογράφησης, η ασύμμετρη κρυπτογράφηση, ή αλλιώς κρυπτογράφηση δημοσίου κλειδιού, δεν βασίζεται στην ύπαρξη ενός συμμετρικού κλειδιού, αλλά σε ένα ζεύγος κλειδιών, το δημόσιο κλειδί, που είναι γνωστό σε όλους και το ιδιωτικό κλειδί που είναι γνωστό μόνο στον χρήστη που δημιουργεί το ζεύγος κλειδιών. Το δημόσιο κλειδί συνήθως προκύπτει από το ιδιωτικό και αξιοποιώντας μαθηματικές συναρτήσεις οι οποίες είναι υπολογιστικά «ελαφριές» όταν γίνεται η παραγωγή του δημοσίου κλειδιού από το ιδιωτικό, ενώ είναι η αντιστροφή τους είναι υπολογιστικά δύσκολη, έως και ακατόρθωτη.

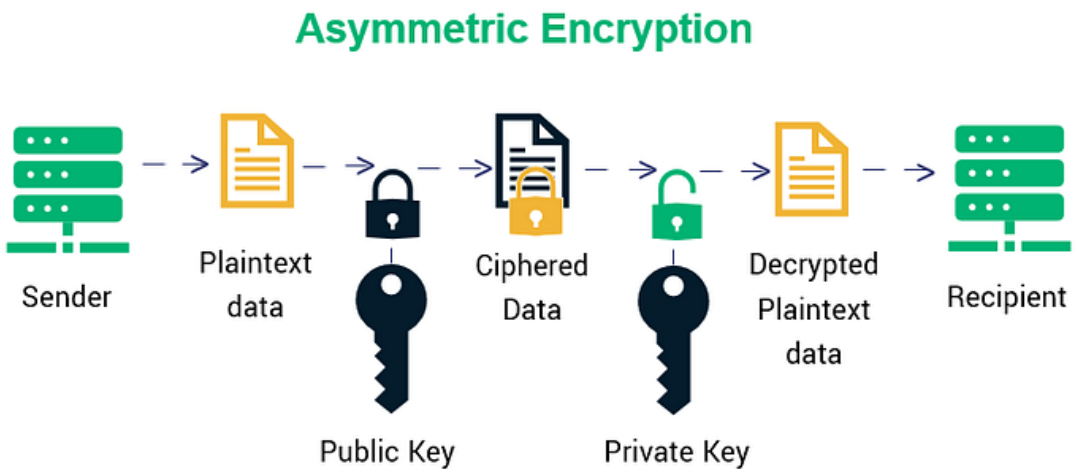
Τη συγκεκριμένη μορφή κρυπτογράφησης την εισήγαγαν οι Diffie, Hellman και Merkle το 1976 και έφεραν επανάσταση στον τομέα της κρυπτογραφίας, αφού άνοιξαν τον δρόμο για την κρυπτογράφηση χωρίς να υπάρχει αρχικά κάποια κατανομή ενός κοινού μυστικού, που απαιτεί η συμμετρική κρυπτογράφηση. Το μεγαλύτερο μέρος της επικοινωνίας στο διαδίκτυο γίνεται πάνω από μη ασφαλή κανάλια και μεταξύ πλευρών που είναι εντελώς «άγνωστες» μεταξύ τους, επομένως ήταν αναγκαία η εύρεση μίας τέτοιας μεθόδου για την διασφάλιση της επικοινωνίας.

Στην ασύμμετρη κρυπτογραφία, στην περίπτωση που η Alice θέλει να στείλει ένα μήνυμα στον Bob πάνω από ένα μη ασφαλές κανάλι, αρχικά θα συμφωνήσει μαζί του για τη χρήση ενός αλγορίθμου ασύμμετρης κρυπτογράφησης. Ο αλγόριθμος είναι αναγκαίο να έχει τα εξής χαρακτηριστικά:

- Να είναι υπολογιστικά εύκολο για τον Bob να παράγει το ζεύγος κλειδιών του.
- Να είναι υπολογιστικά εύκολο για την Alice να κρυπτογραφήσει το αρχικό κείμενο με τη χρήση του δημοσίου κλειδιού του Bob.
- Να είναι υπολογιστικά εύκολο για τον Bob να αποκρυπτογραφήσει το κρυπτογραφημένο κείμενο της Alice με το ιδιωτικό κλειδί του.
- Να είναι υπολογιστικά ανέφικτο για την Trudy, η οποία γνωρίζει τον αλγόριθμο κρυπτογράφησης, δημόσιο κλειδί του Bob και το κρυπτογραφημένο κείμενο, να ανακτήσει το αρχικό κείμενο.

- Να είναι υπολογιστικά αδύνατο για την Trudy, γνωρίζοντας τον αλγόριθμο κρυπτογράφησης και το δημόσιο κλειδί του Bob, να υπολογίσει το ιδιωτικό κλειδί του Bob.

Αφού γίνει η επιλογή τους αλγορίθμου ασύμμετρης κρυπτογράφησης, γίνεται η παραγωγή των κλειδιών από τον Bob, ο οποίος δημοσιεύει το δημόσιο κλειδί του πάνω στο μη ασφαλές κανάλι. Στη συνέχεια, η Alice κρυπτογραφεί το αρχικό κείμενο με το δημόσιο κλειδί του Bob και παράγει το κρυπτογραφημένο κείμενο, το οποίο μοιράζεται με τον Bob πάνω από το μη ασφαλές κανάλι. Τέλος, ο Bob με τη χρήση του ιδιωτικού του κλειδιού και του αλγόριθμου αποκρυπτογράφησης ανακτά το αρχικό κείμενο της Alice [2].



Εικόνα 2: Ασύμμετρη κρυπτογράφηση [2]

Συμπεραίνεται πως το σχήμα της ασύμμετρης κρυπτογράφησης προσφέρει σημαντικά πλεονεκτήματα σε σχέση με αυτό της συμμετρικής. Αρχικά, εξασφαλίζουν την εμπιστευτικότητα στην επικοινωνία, δηλαδή πως μόνο ο παραλήπτης που κατέχει το ιδιωτικό κλειδί θα μπορέσει να αποκρυπτογραφήσει το κείμενο. Πέρα από την κρυπτογράφηση/αποκρυπτογράφηση μηνυμάτων, η κρυπτογράφηση δημοσίου κλειδιού μπορεί να χρησιμοποιηθεί και για την αυθεντικοποίηση αυτών, με τη χρήση της ψηφιακής υπογραφής, οι οποίες αναλύονται παρακάτω. Επιπλέον, το σχήμα μπορεί να δώσει λύσεις στο πρόβλημα της ανταλλαγής του μυστικού κλειδιού που προαναφέρθηκε. Είτε με τη μέθοδο Diffie – Hellman, είτε απλώς με τη κρυπτογράφηση του συμμετρικού κλειδιού με το δημόσιο κλειδί του Bob, η Alice και ο Bob μπορούν να ανταλλάξουν το κοινό μυστικό τους και να συνεχίσουν να επικοινωνούν με συμμετρικό τρόπο [3].

Δύο από τους πιο γνωστούς αλγόριθμους κρυπτογράφησης είναι οι εξής:

- Rivest-Shamir-Adleman (RSA): Ο RSA είναι ένας ευρέως διαδεδομένος αλγόριθμος ασύμμετρης κρυπτογράφησης, ο οποίος βασίζεται στη

μαθηματική δυσκολία παραγοντοποίησης μεγάλων πρώτων αριθμών. Ο RSA προσφέρει μεγάλη ασφάλεια στην κρυπτογράφηση/αποκρυπτογράφηση μηνυμάτων, αλλά και δυνατότητα για ασφαλή ανταλλαγή κλειδιών και ψηφιακής υπογραφής μηνύματος.

- Συστήματα DLP βασισμένα στην Κρυπτογραφία Υπερελλειπτικών Καμπυλών: Το συγκεκριμένο σύστημα αξιοποιεί τη δυσκολία της λύσης του προβλήματος του Διακριτού Λογάριθμου (DLP Problem) [4] για κρυπτογράφηση/αποκρυπτογράφηση μηνυμάτων (μέθοδος ElGamal), ψηφιακές υπογραφές (ECDSA, ElGamal ECC signatures) και ασφαλή ανταλλαγή κλειδιών (ECC Diffie-Hellman). Συγκεκριμένα, περισσότερο διαδομένη και ισχυρή είναι η χρήση των Ελλειπτικών Καμπυλών, που είναι υποκατηγορία των Υπερελλειπτικών Καμπυλών (Υπερελλειπτικές Καμπύλες γένους 1) και προσφέρουν μικρότερα μεγέθη κλειδιών για το ίδιο επίπεδο ασφάλειας με τον RSA .

Παρόλα αυτά, οι αλγόριθμοι ασύμμετρης κρυπτογράφησης έχουν και κάποια αρκετά σημαντικά μειονεκτήματα. Το πιο βασικό είναι το αυξημένο υπολογιστικό κόστος που εισάγουν κατά την κρυπτογράφηση/αποκρυπτογράφηση και στην παραγωγή κλειδιών, το οποίο κάποιες φορές είναι αρκετά μεγαλύτερο από αυτό των συμμετρικών. Γι’ αυτό, συνήθως αυτοί οι αλγόριθμοι χρησιμοποιούνται για την ανταλλαγή ενός συμμετρικού κλειδιού πάνω σε ένα μη ασφαλές κανάλι και στη συνέχεια η επικοινωνία ασφαλίζεται με συμμετρικό τρόπο. Επιπλέον, για να είναι επαρκώς ασφαλείς, οι συγκεκριμένοι αλγόριθμοι συνήθως απαιτούν αρκετά μεγάλα μεγέθη κλειδιών. Για παράδειγμα ο RSA για να επιτύχει επίπεδο ασφαλείας 128-bit, δηλαδή αντίστοιχο επίπεδο ασφαλείας του AES με μήκος κλειδιού 128 bits, απαιτεί μήκος κλειδιού 3072 bits, ενώ ο ECC 256-bits. Εκτός αυτού, η επιλογή των χαρακτηριστικών του ασύμμετρου κρυπτοσυστήματος που θα χρησιμοποιηθεί για τη διασφάλιση της επικοινωνίας πρέπει να γίνεται με ιδιαίτερη προσοχή, ώστε το σύστημα να μην είναι εύκολα παραβιάσιμο με τις γνωστές επιθέσεις, ενώ παράλληλα να είναι αποδοτικό. Επίσης, όπως όλα δείχνουν, οι συγκεκριμένοι αλγόριθμοι μελλοντικά θα είναι πολύ εύκολο να παραβιαστούν με τη χρήση κβαντικών υπολογιστών, αφού ήδη έχουν αναπτυχθεί αλγόριθμοι για τη λύση των προβλημάτων της παραγοντοποίησης μεγάλων πρώτων αριθμών και του προβλήματος του Διακριτού Λογάριθμου με αρκετά αποδοτικό τρόπο.

2.1.3 Ψηφιακές υπογραφές

Όπως προαναφέρθηκε, τα κρυπτοσυστήματα δημοσίου κλειδιού δίνουν τη δυνατότητα για τη δημιουργία ψηφιακών υπογραφών, με σκοπό την αυθεντικοποίηση των μηνυμάτων. Αυτή η λειτουργία είναι απαραίτητη για:

- Την εξασφάλιση της ακεραιότητας των μηνυμάτων, δηλαδή πως αν κάποιος κακόβουλος χρήστης επιχειρήσει να διαστρεβλώσει το περιεχόμενο του μηνύματος για να εξαπατήσει τους χρήστες και να κλέψει δεδομένα, τότε η υπογραφή θα είναι άκυρη.
- Την αυθεντικοποίηση των μηνυμάτων, δηλαδή την εξασφάλιση από τον λαμβάνων πως το μήνυμα προήλθε πράγματι από την αναμενόμενη πηγή και όχι από κάποιον κακόβουλο χρήστη που προσπαθεί να υποδυθεί τον αποστολέα για να κλέψει μηνύματα.
- Την αποφυγή άρνησης του αποστολέα ότι έστειλε κάποιο μήνυμα. Αυτό είναι ιδιαίτερα σημαντικό για περιπτώσεις νομικής και επιχειρηματικής φύσεως, ώστε να εξασφαλίζεται πως αφού ο αποστολέας υπέγραψε το μήνυμα, δεν μπορεί αργότερα να αρνηθεί την πράξη του.

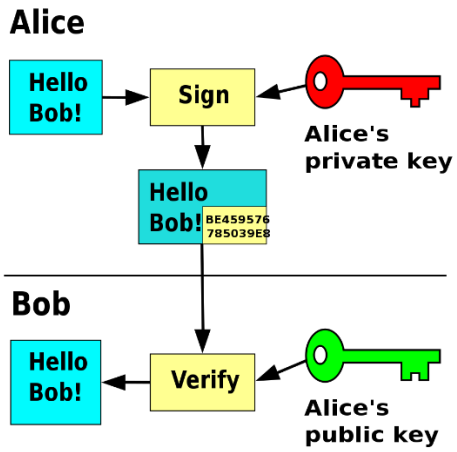
Με βάση τα παραπάνω συμπεραίνεται πως η ψηφιακή υπογραφή πρέπει να καλύπτει τις εξής προϋποθέσεις:

- Η υπογραφεί του μηνύματος πρέπει να γίνεται από μία μυστική πληροφορία που κατέχει μόνο ο αποστολέας.
- Η επικύρωση του μηνύματος πρέπει να μπορεί να γίνει εύκολα και μόνο με τη χρήση της δημόσιας πληροφορίας που μοιράζεται ο αποστολέας, η οποία έχει προκύψει με κάποιο μαθηματικό τέχνασμα από την μυστική πληροφορία.
- Να είναι σχεδόν ανέφικτη η επικύρωση δύο διαφορετικών υπογραφών που παράχθηκαν με διαφορετικές μυστικές πληροφορίες, με την δημόσια πληροφορία.
- Η παραγωγή της υπογραφής και η επικύρωσή της πρέπει να γίνονται υπολογιστικά εύκολα.

Επομένως, η χρήση ενός κρυπτογραφικού συστήματος δημοσίου κλειδιού ταιριάζει απόλυτα στις απαιτήσεις του σχήματος ψηφιακής υπογραφής [5].

Συνεχίζοντας το προηγούμενο παράδειγμα της Alice και του Bob, έστω ότι η Alice επιθυμεί να στείλει ένα μήνυμα στον Bob πάνω από ένα μη ασφαλές κανάλι. Η Trudy, η οποία προσπαθεί να κλέψει την πληροφορία, έχει την επιλογή να αντικαταστήσει το μήνυμα της Alice με ένα δικό της, ώστε να παραπλανήσει τον Bob και να επέμβει στην επικοινωνία τους. Για να αποφευχθεί αυτό, η Alice κάνει χρήση της ψηφιακής υπογραφής της. Δηλαδή, υπογράφει το μήνυμα που επιθυμεί να στείλει στον Bob με το μυστικό κλειδί που έχει παράγει και δημοσιεύει το δημόσιο κλειδί της στον Bob. Μαζί με το κρυπτογραφημένο μήνυμα, η Alice στέλνει και την ψηφιακή υπογραφή, την οποία ο Bob επικυρώνει με το δημόσιο κλειδί της Alice. Έτσι,

ο Bob επιβεβαιώνει πως το μήνυμα πράγματι έχει προέλθει από την Alice και δεν έχει υποστεί καμία αλλοίωση.

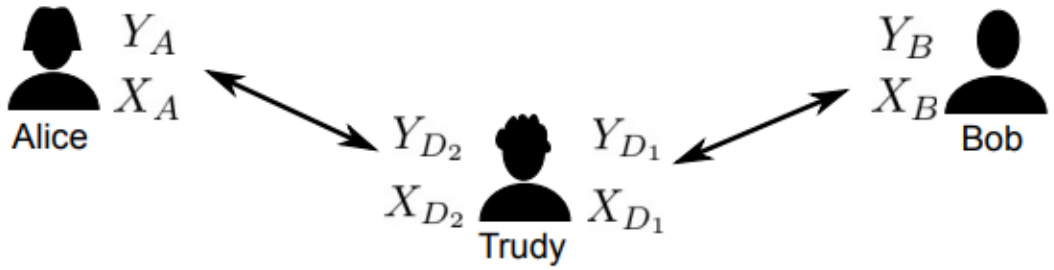


Εικόνα 3: Ψηφιακή υπογραφή [5]

Το παραπάνω σχήμα, επομένως, αντιμετωπίζει τα ζητήματα ακεραιότητας και αυθεντικοποίησης των μηνυμάτων. Ωστόσο, η Trudy έχει τη δυνατότητα να αντικαταστήσει το δημόσιο κλειδί της Alice με ένα δικό της και να το στείλει στον Bob, επεμβαίνοντας έτσι στην επικοινωνία υποδυόμενη την Alice. Για αυτό, είναι απαραίτητο να υπάρχει και ένα σχήμα ταυτοποίησης των δημοσίων κλειδιών. Αυτή η διαδικασία επιτυγχάνεται με τη χρήση των ψηφιακών πιστοποιητικών δημοσίου κλειδιού, τα οποία αναλύονται παρακάτω.

2.1.4 Πιστοποιητικά δημοσίου κλειδιού

Η επίθεση που αναφέρθηκε παραπάνω, είναι γνωστή ως επίθεση Man-in-the-Middle [6] και είναι ένα από τα κυριότερα ζητήματα ασφαλείας στα δίκτυα επικοινωνίας σήμερα. Συγκεκριμένα, στην επικοινωνία μεταξύ της Alice και του Bob, η Trudy έχει τη δυνατότητα να παρέμβει στην επικοινωνία κατά τη διάρκεια ανταλλαγής των δημοσίων κλειδιών και να στείλει στον Bob και στην Alice ένα ή περισσότερα δικά της δημόσια κλειδιά. Έτσι η επικοινωνία εγκαθίσταται μεταξύ της Trudy και του Bob και της Alice, με αποτέλεσμα οι δύο τελευταίοι να θεωρούν πως επικοινωνούν μεταξύ τους, αλλά στην πραγματικότητα να κρυπτογραφούν και να υπογράφουν με τα στοιχεία της Trudy, η οποία πλέον έχει τη δυνατότητα να ανακτήσει όλη την πληροφορία. Η Trudy μεταβιβάζει τα μηνύματα από την μία πλευρά στην άλλη, αλλάζοντας τις υπογραφές με τις δικές της με αποτέλεσμα οι δύο πλευρές να μην αντιλαμβάνονται πως υπάρχει κάποιος κακόβουλος χρήστης που «κρυφακούει». Όλα τα κρυπτοσυστήματα δημοσίου κλειδιού είναι εκτεθειμένα σε αυτή την επίθεση.



Εικόνα 4: Επίθεση Man-in-the-Middle (Από: μάθημα «Ασφάλεια Δικτύων Υπολογιστών 8ου εξαμήνου ΗΜΜΥ ΕΜΠ διαφάνειες Μη συμμετρικής κρυπτογραφίας σελ. 19)

Για να αντιμετωπιστεί με αποτελεσματικότητα η παραπάνω επίθεση, είναι απαραίτητη η ύπαρξη ενός τρόπου επικύρωσης των δημοσίων κλειδιών, ώστε η Alice και ο Bob να μπορούν να σιγουρευτούν πως τα κλειδιά που λαμβάνουν δεν προέρχονται από κάποια κακόβουλη πηγή. Με σκοπό την εξασφάλιση της αυθεντικότητας των δημοσίων κλειδιών, δημιουργήθηκε η έννοια των πιστοποιητικών δημοσίου κλειδιού [7]. Αρχικά, για την ορθή λειτουργία των πιστοποιητικών είναι απαραίτητη η θεμελίωση μίας επίσημης Αρχής Πιστοποιητικών (Certificate Authority – CA), η οποία αναλαμβάνει να εκδίδει, να συντηρεί και να ανανεώνει πιστοποιητικά για τους χρήστες. Οι Αρχές Πιστοποιητικών δεσμεύονται για την ακεραιότητα των πιστοποιητικών που διαδίδουν και από την άλλη οι χρήστες εμπιστεύονται την Αρχή Πιστοποιητικών για να επικυρώνουν ότι ένας χρήστης είναι νόμιμος. Ένα πιστοποιητικό συνήθως περιέχει:

- Πεδίο θέματος: Αν το δημόσιο κλειδί που διαδίδει το παρόν πιστοποιητικό αφορά ιδιώτη, οργανισμό ή συσκευή.
- Το δημόσιο κλειδί που διαδίδεται, είτε αυτούσιο, είτε με τη μορφή μιας πληροφορίας, η οποία σε συνδυασμό με ένα χαρακτηριστικό της Αρχής Πιστοποιητικών μπορεί να παράγει το επίσημο δημόσιο κλειδί.
- Το όνομα ή αναγνωριστικό της Αρχής Πιστοποιητικών που παρήγαγε και διένειμε στον εκάστοτε χρήστη.
- Έναν μοναδικό σειριακό αριθμό για κάθε πιστοποιητικό.
- Ψηφιακή υπογραφή. Η υπογραφή παράγεται από την Αρχή Πιστοποιητικών, η οποία υπογράφει το δημόσιο κλειδί που θα διανεμηθεί με ένα μυστικό κλειδί της. Για να επικυρωθεί πως το δημόσιο κλειδί είναι εμπιστεύσιμο, ο χρήστης που το λαμβάνει επικυρώνει την υπογραφή με το δημόσιο κλειδί της Αρχής Πιστοποιητικών.

Πέρα από τα παραπάνω είναι αναγκαία η ύπαρξη ενός μηχανισμού που θα ενημερώνει τους χρήστες για πλέον άκυρα πιστοποιητικά και να τα ανακαλεί, όταν για παράδειγμα έχει λήξει η περίοδος εγκυρότητάς του ή όταν κάποιος χρήστης που κατέχει πιστοποιητικό διαπιστωθεί ότι έχει κακόβουλη συμπεριφορά. Με σκοπό την επίτευξη του παραπάνω στόχου, δημιουργήθηκαν οι Λίστες Ανάκλησης Πιστοποιητικών, ή αλλιώς Certificate Revocation List – CRL. Η διανομή της λίστας σε

όλους τους χρήστες, ωστόσο, επιβαρύνει την επικοινωνία, καθώς κάθε χρήστης θα πρέπει να ανανεώνει συχνά τη λίστα και να κρατάει ένα αντίγραφό της αποθηκευμένο. Οι πιο συχνές προσεγγίσεις είναι η Αρχή Πιστοποιητικών να ανανεώνει σε συχνή βάση τη Λίστα και να τη στέλνει στους χρήστες, ωστόσο υπάρχει κίνδυνος κάποιος κακόβουλος χρήστης να συνεχίσει τη δράση του έως ότου η Λίστα διανεμηθεί εκ νέου. Από την άλλη, ο χρήστης μπορεί να επικοινωνεί με την Αρχή κάθε φορά που λαμβάνει ένα πιστοποιητικό, πράγμα το οποίο δημιουργεί επιβάρυνση στο δίκτυο.

Εν κατακλείδι, η ασύμμετρη κρυπτογραφία δημιούργησε πολλές νέες δυνατότητες για την διασφάλιση της ιδιωτικότητας στα δίκτυα επικοινωνίας και πλέον αποτελεί αναπόσπαστο κομμάτι της καθημερινότητας, μιας και μπορεί να καλύψει τις ανάγκες για κρυπτογράφηση, αυθεντικοποίηση και εγκυρότητα. Από τους πιο γνωστούς αλγόριθμους που καλύπτουν τις παραπάνω ανάγκες είναι τα κρυπτογραφικά συστήματα των Υπερελλειπτικών Καμπυλών, τα οποία αξιοποιούνται στην παρούσα εργασία. Είναι σημαντικό να γίνει μία σύντομη παρουσίαση των εννοιών τους για την ευρύτερη κατανόηση της έρευνας που έγινε.

2.2 Κρυπτογραφικά στοιχεία υπερελλειπτικών καμπυλών

2.2.1 Θεωρία των ομάδων

Η θεωρία των Ομάδων βασίζεται στη θεωρία των αριθμών, τη θεωρία των αλγεβρικών εξισώσεων και τη γεωμετρία. Ο άνθρωπος που θέσπισε τον όρο Ομάδα, ονομάζεται Évariste Galois, ο οποίος ήταν Γάλλος μαθηματικός που έζησε στις αρχές του 19^{ου} αιώνα. Η Θεωρία των Ομάδων πλέον έχει πολλές πρακτικές εφαρμογές σε διάφορα επιστημονικά πεδία και ειδικά σε αυτό της κρυπτογραφίας, στο οποίο εστιάζει η παρούσα εργασία.

Ως Ομάδα [8], στα μαθηματικά και στην αφηρημένη Άλγεβρα ορίζεται ως ένα ζεύγος $(G, *)$, όπου G είναι ένα σύνολο και $*$ είναι μία δυαδική πράξη τέτοια ώστε $*: G \times G \to G, (a, b) \mapsto a * b$ και επίσης:

- ισχύει η προσεταιριστική ιδιότητα, δηλαδή για κάθε τριάδα στοιχείων της ομάδας a, b, c ισχύει $(a * b) * c = a * (b * c)$,
- υπάρχει ουδέτερο στοιχείο, δηλαδή υπάρχει ένα στοιχείο e στην ομάδα τέτοιο ώστε για κάθε στοιχείο a της ομάδας ισχύει: $a * e = e * a = a$,
- υπάρχει αντίστροφο στοιχείο, δηλαδή για κάθε στοιχείο a της ομάδας υπάρχει ένα στοιχείο $a^{-1} \in G$ τέτοιο ώστε $a * a^{-1} = a^{-1} * a = e$.

Υπάρχουν διάφορα είδη Ομάδων, καθώς η Ομάδα σαν έννοια από μόνη της είναι αφηρημένη και έτσι η επέκταση του ορισμού τους με περισσότερες ιδιότητες δημιουργεί διάφορες οικογένειες ομάδων. Κάποιες από αυτές είναι οι Ομάδες Μεταθέσεων, οι Ομάδες Πινάκων και οι Ομάδες Μετασχηματιστών. Μία από τις

οικογένειες που έχει ιδιαίτερο ενδιαφέρον στην κρυπτογραφία είναι οι Κυκλικές Ομάδες [9]. Οι Κυκλικές Ομάδες είναι οι ομάδες των οποίων τα στοιχεία είναι οι ακέραιες δυνάμεις ενός αρχικού στοιχείου g της ομάδας, που ονομάζεται γεννήτρια του G. Η Κυκλική Ομάδα ορίζεται ως $\langle g \rangle = \{ g^k \mid k \in \mathbb{Z} \}$. Η τάξη (order) του g ορίζεται ως $|\langle g \rangle|$ και είναι ο αριθμός των στοιχείων που απαρτίζουν το $\langle g \rangle$. Μία Πεπερασμένη Κυκλική ομάδα τάξης n περιλαμβάνει όλες τις δυνάμεις του g από το 0 έως το $n-1$ και το ουδέτερο στοιχείο. Αυτές οι Ομάδες είναι ισομορφικές ως προς την Ομάδα των Ακέραιων αριθμών υπόλοιπο n.

Η Ομάδα των Ακέραιων αριθμών υπόλοιπο n είναι ένα από τα πιο διαδεδομένα παραδείγματα ενός Πεπερασμένου Σώματος (Finite Fields) [10], μία άλλη αλγεβρική έννοια που εισηγήθηκε ο Galois, γι' αυτό και αυτά τα Σώματα ονομάζονται και Σώματα Galois (Galois Fields ή GF). Τα Πεπερασμένα Σώματα τάξης q, όπου το q είναι δύναμη ενός πρώτου αριθμού, δημιουργούν μία Πεπερασμένη Κυκλική Ομάδα. Ένα από τα πιο απλά παραδείγματα ενός τέτοιο Πεπερασμένου Σώματος είναι οι Ακέραιοι Αριθμοί υπόλοιπο p, όπου το p είναι πρώτος αριθμός ($\mathbb{Z}/p\mathbb{Z}$) . Αυτό το Πεπερασμένο σώμα είναι τάξης p και δημιουργεί μία Κυκλική Ομάδα, πράγμα που είναι πολύ χρήσιμο στην κρυπτογραφία.

Οι κρυπτογραφικές τεχνικές που βασίζονται στο πρόβλημα του Διακριτού Λογάριθμου αξιοποιούν τις ιδιότητες των Κυκλικών Ομάδων για να δημιουργήσουν μία διαδικασία, η οποία δεν είναι εύκολα αντιστρέψιμη και να ορίσουν ένα κρυπτογραφικό σύστημα δημοσίου κλειδιού. Ο τρόπος με τον οποίο γίνεται αυτό εξηγείται παρακάτω.

2.2.2 Το πρόβλημα του διακριτού λογάριθμου

Το πρόβλημα του Διακριτού Λογάριθμου [4] μπορεί να οριστεί ως εξής: Έστω οποιαδήποτε Ομάδα G και έστω ότι η πράξη που την ορίζει είναι ο πολλαπλασιασμός και το ουδέτερο στοιχείο του να είναι το 1. Για οποιαδήποτε δύο στοιχεία a και b, ο ακέραιος αριθμός k που λύνει την εξίσωση $b^k = a$, ονομάζεται διακριτός λογάριθμος του a με βάση b και ορίζεται ως $k = log_b a$.

Το πρόβλημα του Διακριτού Λογάριθμου στο πεδίο των ακέραιων αριθμών είναι ένα σχετικά εύκολο πρόβλημα και επομένως δεν έχει ιδιαίτερη κρυπτογραφική σημασία. Από την άλλη, στις Κυκλικές Ομάδες, όπου κάθε στοιχείο μπορεί να εκφραστεί ως μια δύναμη της γεννήτριας, ο Διακριτός Λογάριθμος γίνεται αρκετά δύσκολος. Για παράδειγμα, στην Ομάδα των Ακέραιων αριθμών υπόλοιπο p, κάθε αριθμός στο $[1, p-1]$ μπορεί να εκφραστεί ως δύναμη του στοιχείου γεννήτριας, έστω a. Δηλαδή, υψώνοντας τη γεννήτρια a σε ακέραιες δυνάμεις $k \in [1, p-1]$ το αποτέλεσμα είναι να παραχθούν όλοι οι αριθμοί από το 1 έως και το $p-1$ με «σχετικά» τυχαίο τρόπο, ακολουθώντας μία ομοιόμορφη κατανομή. Οι καλύτεροι αλγόριθμοι που λύνουν το πρόβλημα στις Ομάδες Ακέραιων αριθμών υπόλοιπο p είναι στην καλύτερη περίπτωση υπό-εκθετικοί (sub-exponential) και δεν υπάρχει

αλγόριθμος που να το λύνει σε γραμμικό χρόνο. Με κατάλληλη επιλογή του πρώτου αριθμού p, ώστε να είναι αρκετά μεγάλος, και της γεννήτριας, ώστε να δημιουργεί μία Κυκλική Ομάδα τάξης p το πρόβλημα είναι ιδιαίτερα αργό να λυθεί στα τωρινά υπολογιστικά συστήματα, πράγμα που το καθιστά κατάλληλο για την ανάπτυξη κρυπτογραφίας δημοσίου κλειδιού βασισμένη σε αυτό.

Ωστόσο, ο αλγόριθμος γνωστός ως Number Field Sieve, επιτυγχάνει να λύσει το πρόβλημα του Διακριτού Λογάριθμου στο $\mathbb{Z}/p\mathbb{Z}$ για μερικές εκατοντάδες bits σε χρόνο, ο οποίος είναι εφαρμόσιμος. Αυτό δημιουργεί την απαίτηση ο πρώτος αριθμός p να είναι πολύ μεγάλος, συνήθως τουλάχιστον μεγέθους 1024-bit για να είναι εγγυημένη η ασφάλεια του συστήματος. Ο αλγόριθμος αξιοποιεί κάποιες ιδιότητες των αριθμών που αναλύονται αποκλειστικά σε μικρούς πρώτους παράγοντες. Επομένως, δημιουργείται η ανάγκη για τη θέσπισης μίας άλλης Κυκλικής Ομάδας με παρόμοιες ιδιότητες, στην οποία όμως ο Number Field Sieve δεν μπορεί να έχει εφαρμογή. Έτσι γεννήθηκε η ιδέα για τη χρήση ελλειπτικών/υπερελλειπτικών καμπυλών για τη δημιουργία μίας αντίστοιχης Κυκλικής Ομάδας.

2.2.3 Κρυπτογραφία ελλειπτικών καμπυλών

Οι μαθηματικοί Koblitz και Miller ξεχωριστά πρότειναν το 1985 την Ομάδα των σημείων μίας ελλειπτικής καμπύλης που ορίζεται πάνω σε ένα Πεπερασμένο Σώμα, για την αξιοποίηση του Προβλήματος του Διακριτού Λογάριθμου στην κρυπτογραφία. Καθώς στα σημεία της ελλειπτικής καμπύλης δεν μπορεί να αξιοποιηθεί η ιδιότητα των πρώτων παραγόντων που χρησιμοποιεί ο αλγόριθμος Number Field Sieve που προαναφέρθηκε, η καλύτερη μέθοδος επίλυσης του Διακριτού Λογάριθμου στην Ομάδα είναι πολυπλοκότητας τετραγωνικής ρίζας, πράγμα που καθιστά την Ομάδα κατάλληλη για κρυπτογραφικές τεχνικές [11].

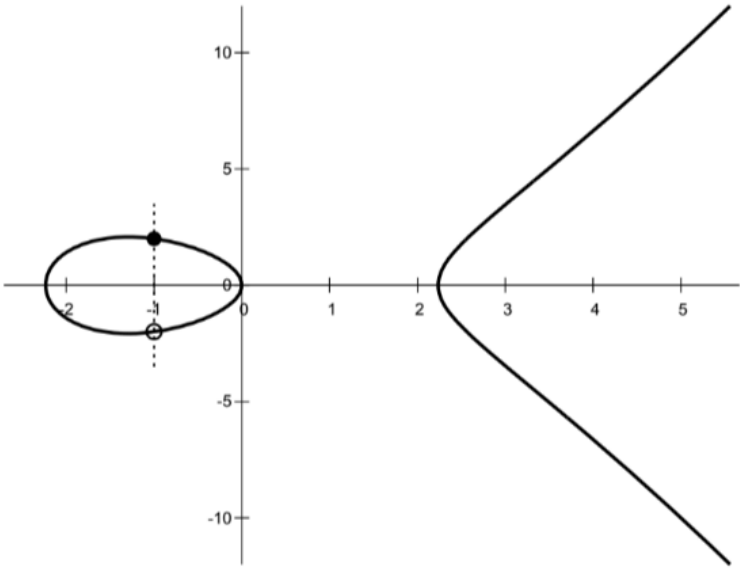
Για να χρησιμοποιηθούν οι καμπύλες για τη δημιουργία της επιθυμητής ομάδας πρέπει, αρχικά, να οριστούν τα στοιχεία της Ομάδας, καθώς και η πράξη που θα χρησιμοποιηθεί για να ορίσει την Ομάδα. Η καμπύλη και επαγωγικά τα σημεία που ορίζουν την Ομάδα δίνονται από την εξίσωση:

$$E: y^2 = x^3 + Ax + B, \qquad (A,B \in \mathbb{K}) \qquad (2.1)$$

Επίσης, πρέπει να ισχύει για την καμπύλη ότι είναι non-singular, δηλαδή η διακρίνουσα $-(4A^3 + 27B^3)$ να μην εξαφανίζεται και το Πεπερασμένο Σώμα $\mathbb{K}$ να έχει χαρακτηριστικό διαφορετικό από 2 και 3. Επομένως, τα στοιχεία της ομάδας αποτελούνται από τα σημεία που ορίζουν την καμπύλη, μαζί με ένα ουδέτερο στοιχείο, το σημείο στο άπειρο.

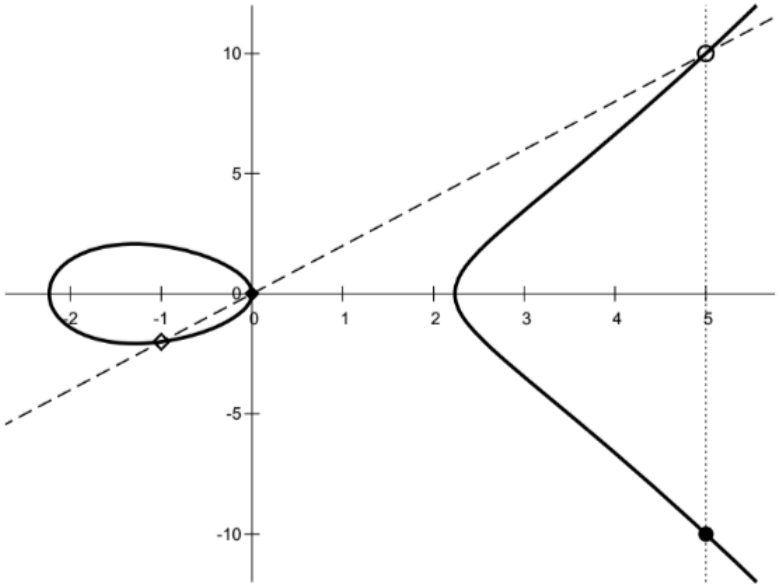
Τώρα, πρέπει να οριστεί η πράξη της Ομάδας. Παρατηρείται πως στη συγκεκριμένη μορφή των καμπυλών κάθε σημείο P = (x_0, y_0), έχει ένα αντίστροφο

σημείο, το οποίο είναι το 2ο σημείο που η ευθεία $x = x_0$ τέμνει την καμπύλη, δηλαδή στο $P' = (x_0, -y_0)$.



Εικόνα 5: Αντίστροφο σημείο Ελλειπτικής Καμπύλης [11]

Για να οριστεί η πράξη τη Ομάδας, χρησιμοποιούνται δύο σημεία της καμπύλης, έστω P και Q. Η ευθεία που διέρχεται από τα δύο σημεία, τέμνει την καμπύλη σε ένα τρίτο σημείο το R. Επομένως μπορεί να οριστεί η πράξη $P \oplus Q$, η οποία δίνει ως αποτέλεσμα το σημείο R' (το αντίστροφο του σημείου R), αφού τα P, Q και R είναι συνευθειακά.



Εικόνα 6: Κανόνας Ομάδας σημείων Ελλειπτικής Καμπύλης [11]

Για την αξιοποίηση του Διακριτού Λογάριθμου, ορίζεται ο βαθμωτός πολλαπλασιασμός ως:

$$P \mapsto [n]P = P \oplus P \oplus P \ldots. \oplus P \qquad \text{για n φορές.}$$

Έτσι, γίνεται εμφανής η παρουσία του προβλήματος του Διακριτού Λογάριθμου όπου P είναι το στοιχείο γεννήτρια και n είναι η δύναμη στην οποία υψώνεται η γεννήτρια και δημιουργεί την Κυκλική Ομάδα, με τα επιθυμητά χαρακτηριστικά της ομοιόμορφης κατανομής.

Επομένως, στα κρυπτογραφικά συστήματα με βάση τις Ελλειπτικές Καμπύλες (ECC), το στοιχείο n που πολλαπλασιάζει βαθμωτά τη γεννήτρια P μπορεί να θεωρηθεί ως το μυστικό κλειδί του χρήστη, ενώ το αποτέλεσμα της πράξης είναι το δημόσιο κλειδί. Για να μπορέσει κάποιος επιτιθέμενος να ανακτήσει το μυστικό κλειδί από το δημόσιο θα πρέπει να λύσει το πρόβλημα του Διακριτού Λογάριθμου στην Κυκλική Ομάδα που ορίστηκε παραπάνω, πράγμα που είναι ιδιαίτερα δύσκολο. Από την άλλη, η πράξη του βαθμωτού πολλαπλασιασμού είναι σχετικά εύκολη και με διάφορα μαθηματικά τεχνάσματα μπορεί να υπολογιστεί πολύ αποδοτικά. Από την άλλη, είναι σημαντικό η γεννήτρια και οι παράμετροι της καμπύλης να επιλέγονται με προσοχή, ώστε να παράγεται μία Κυκλική Ομάδα που έχει τάξη (order) έναν πρώτο αριθμό. Αυτό είναι ένα σχετικά δύσκολο πρόβλημα, ωστόσο στις ελλειπτικές καμπύλες υπάρχουν πλέον αποδοτικοί τρόποι να παραχθούν ασφαλείς παράμετροι καμπυλών και γεννητριών. Αποδεικνύεται πως η τάξη της Κυκλικής Ομάδας είναι περίπου $p \pm O(\sqrt{p})$ [11], όπου το p είναι ο πρώτος αριθμός που έχει επιλεγεί για να ορίσει το Πεπερασμένο Σώμα, επομένως μπορεί να γίνει εκτίμηση της ασφάλειας που δίνει κάθε καμπύλη. Συγκεκριμένα, εφόσον οι καλύτεροι αλγόριθμοι που λύνουν τον Διακριτό Λογάριθμο έχουν πολυπλοκότητα $O(\sqrt{p})$ και το p έστω ότι έχει μήκος n bits, τότε για την εξασφάλιση ενός επιπέδου ασφαλείας 128-bit, το p αρκεί να είναι μεγέθους n/2 ίσο με 128, δηλαδή 256 bits.

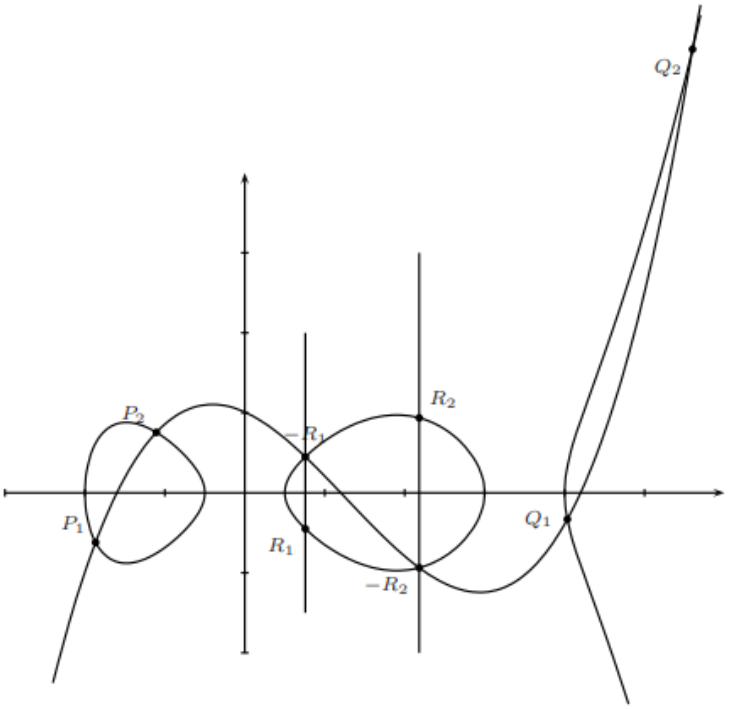
Η κρυπτογραφία Ελλειπτικών Καμπυλών είναι ευρέως διαδεδομένη σήμερα, με εφαρμογές στο πεδίο των κρυπτονομισμάτων, στην εξασφάλιση της ασφάλειας σε συστήματα με περιορισμένη υπολογιστική δύναμη και μνήμη, όπως τα ενσωματωμένα συστήματα και στο Διαδίκτυο των Αντικειμένων, σε ψηφιακά πιστοποιητικά και υπογραφές. Έχει πραγματοποιηθεί εκτενής έρευνα στο συγκεκριμένο αντικείμενο και υπάρχει επίσημη βάση δεδομένων που εμπεριέχει ασφαλείς παραμέτρους για ορισμό Καμπυλών σε κρυπτογραφικά συστήματα, καθώς και πολλές βελτιώσεις στους υπολογισμούς που απαιτούνται, με αποτέλεσμα να είναι εύκολα εφαρμόσιμες σε εφαρμογές χωρίς την ανάγκη για γνώση των μαθηματικών εννοιών που τις ορίζουν. Επιπλέον, παρέχουν ασφάλεια με αρκετά μικρά μήκη κλειδιών, συγκριτικά με τον RSA.

Ωστόσο, η προσεκτική επιλογή των παραμέτρων που απαιτείται μπορεί να δημιουργήσει και κενά ασφαλείας στο σχήμα που θα επιλεγεί για μία εφαρμογή, αν ο μηχανικός που το επιλέγει δεν γνωρίζει πλήρως το μαθηματικό υπόβαθρο και κάνει κάποιο λάθος στην επιλογή. Επιπλέον, το Πρόβλημα του Διακριτού Λογάριθμου δεν είναι ασφαλές από κβαντικές επιθέσεις και επομένως, μελλοντικά, μπορεί οι Ελλειπτικές Καμπύλες να μην επαρκούν για να εξασφαλίσουν την ασφάλεια στα σύγχρονα δίκτυα.

2.2.4 Κρυπτογραφία υπερελλειπτικών καμπυλών γένους ≥ 2

Πέντε χρόνια μετά την πρότασή του για τις Ελλειπτικές Καμπύλες, ο Koblitz πρότεινε τη χρήση της Ιακωβιανής των Υπερελλειπτικών Καμπυλών για τη δημιουργία μίας κατάλληλης Ομάδας. Εξ' άλλου οι Ελλειπτικές Καμπύλες είναι μία υποκατηγορία των Υπερελλειπτικών Καμπυλών, αφού ουσιαστικά είναι η οικογένεια των Υπερελλειπτικών Καμπυλών γένους 1.

Για να ακολουθηθεί η αντίστοιχη διαδικασία θέσπισης μίας Κυκλικής Ομάδας με βάση τις Υπερελλειπτικές Καμπύλες γένους ≥ 2 δεν αρκεί απλώς να θεωρήσουμε σαν πράξη της ομάδας την διαδικασία που περιγράφηκε προηγουμένως, καθώς τώρα κάθε ευθεία μπορεί να τέμνει την καμπύλη έως και σε 2*g+1 σημεία, όπου g είναι το γένος της καμπύλης [11].



Εικόνα 7: Υπερελλειπτική Καμπύλη γένους 2 με $y^2 = f(x)$ [11]

Η εξίσωση μίας Υπερελλειπτικής Καμπύλης γένους g δίνεται από την παρακάτω εξίσωση:

$$C: y^2 + h(x)y = f(x),\ h, f \in K[x], deg(f) = 2g + 1,\ deg(h) \leq g, f\ \mu o \nu \iota \kappa \acute{o}$$

Όπως πριν, πρέπει η καμπύλη να είναι non-singular και αυτό εξασφαλίζεται με τη συνθήκη πως κανένα σημείο δεν μηδενίζει και τις δύο μερικές παραγώγους της.

Για να δημιουργηθεί μία Κυκλική Ομάδα, η μέθοδος που προτάθηκε είναι να χρησιμοποιηθεί ένα σύνολο σημείων που το άθροισμά τους ακολουθεί μια συνάρτηση. Για παράδειγμα, σε μία καμπύλη όπως της εικόνας 6, η οποία είναι καμπύλη γένους 2, τα σημεία $R_1 = (x_{R1}, y_{R1})$ και $-R_1 = (x_{R1}, -y_{R1})$ ανήκουν στην καμπύλη $x = x_{R1}$ και επομένως μπορεί να θεωρηθεί ότι $R_1 \oplus (-R_1) = 0$. Αντίστοιχα, τα σημεία P_1, P_2, Q_1, Q_2, $-R_1$, $-R_2$, ακολουθούν μία κυβική συνάρτηση και άρα αθροίζονται στο

μηδέν. Επομένως, μπορεί ως στοιχεία μπορούν να θεωρηθούν τα στοιχεία που προκύπτουν από το άθροισμα δύο (γενικά g σημείων, όπου g το γένος) σημείων και η πράξη μεταξύ δύο (ή γενικότερα g) στοιχείων δίνει ως αποτέλεσμα τα δύο επιπλέον σημεία που μία κυβική (ή γενικότερα τάξης $g+1$) συνάρτηση $y = s(x)$ τέμνει την Υπερελλειπτική καμπύλη.

Η Ομάδα που περιγράφηκε παραπάνω ονομάζεται η Ομάδα της Κλάσης των Divisor Pic_C^0 της Καμπύλης. Για να οριστεί με σωστό τρόπο η Ομάδα συμπεριλαμβάνεται ξανά το σημείο στο άπειρο, το οποίο πρακτικά θεωρείται ότι τέμνει στο άπειρο κάθε ευθεία παράλληλη με τον άξονα y. Η Κλάση των Divisor παίρνει διάφορες μορφές, ωστόσο αυτή που επικρατεί για τον ορισμό της Ομάδας και για την ανάπτυξη της αριθμητικής της είναι η αναπαράσταση Mumford. Σε αυτή την αναπαράσταση τα στοιχεία της Ομάδας είναι δύο πολυώνυμα $u(x)$ και $v(x)$ τα οποία ακολουθούν τα εξής χαρακτηριστικά:

- το u είναι μονικό
- $\deg(v) < \deg(u) \leq g$ (γένος)
- το u διαιρεί ακριβώς το πολυώνυμο $v^2 + vh - f$

Επομένως, το πολυώνυμο u(x) μπορεί να εκφραστεί ως:

$$u(x) = \prod_{i=1}^{r}(x - x_i)$$

όπου x_i η x συντεταγμένη των σημείων που επιλέγονται να διαμορφώσουν το στοιχείο. Συγκεκριμένα, για τα P_i σημεία που συμμετέχουν (σημεία υποστήριξης) πρέπει να ισχύει ότι $Pi \neq P_\infty$, $P_i \neq -P_j$ για κάθε $i \neq j$. Η συγκεκριμένη μορφή απευθύνεται στην κλάση των Reduced Divisors, όταν ισχύει ότι $r \leq g$. Αφού ορίστηκε, λοιπόν, η Κλάση η οποία πλαισιώνει την Κυκλική Ομάδα ενδιαφέροντος είναι σημαντικό να αναπτυχθούν και αποδοτικοί τρόποι για τον υπολογισμό της πράξης της Ομάδας που περιγράφηκε παραπάνω. Διάφοροι αλγόριθμοι έχουν υλοποιηθεί και βελτιστοποιηθεί για την εκτέλεση αυτής της πράξης, ωστόσο, όπως εύκολα φαίνεται είναι πιο δύσκολη υπολογιστικά από αυτή που ορίστηκε για τις Ελλειπτικές Καμπύλες. Το γεγονός που κάνει τις Υπερελλειπτικές Καμπύλες υποσχόμενες στην κρυπτογραφία, παρόλα αυτά, είναι πως για να επιτευχθεί ένα απαιτούμενο επίπεδο ασφαλείας, το Πεπερασμένο Σώμα που απαιτείται είναι μικρότερο από αυτό που απαιτείται στις Ελλειπτικές Καμπύλες και επομένως η αριθμητική που απαιτείται γίνεται σε μικρότερους αριθμούς.

Αυτό συμβαίνει, γιατί, εφόσον τα στοιχεία της Ομάδας πλέον αποτελούνται από g σημεία και όχι από ένα, η τάξη της έχει μέγεθος p^g, όταν το p είναι αρκετά μεγάλο. Επομένως, με βάση τον υπολογισμό του επιπέδου ασφαλείας που σχολιάστηκε στις Ελλειπτικές Καμπύλες, μία Υπερελλειπτική Καμπύλη γένους 2 μπορεί να επιτύχει επίπεδο ασφαλείας 128-bit με Πεπερασμένο Σώμα μεγέθους 128-bit, ενώ μία Υπερελλειπτική Καμπύλη γένους 3 με Πεπερασμένο Σώμα μεγέθους 86 bit. Παρατηρείται πως όσο αυξάνεται το γένος, το μέγεθος του Πεπερασμένου Σώματος μειώνεται και επομένως τα κλειδιά που θα χρησιμοποιηθούν μπορούν να είναι μικρότερα και να επιτυγχάνουν το ίδιο επίπεδο ασφαλείας. Δυστυχώς, όμως, η

επίθεση Index Calculus, που δεν είναι δυνατή στις Ελλειπτικές Καμπύλες, στις Υπερελλειπτικές Καμπύλες μπορεί να λύσει το Πρόβλημα του Διακριτού Λογάριθμου σε ικανοποιητικό χρόνο, όταν το γένος αυξάνεται. Για αυτό τον λόγο οι καμπύλες γένους μεγαλύτερου ή ίσου με 4 δεν χρησιμοποιούνται στην κρυπτογραφία [12]. Οι καμπύλες γένους 2 είναι αρκετά ασφαλής, ενώ οι καμπύλες γένους 3 χρησιμοποιούνται αλλά δεν είναι όσο ασφαλής όσο οι Ελλειπτικές Καμπύλες.

Η κρυπτογραφία Υπερελλειπτικών Καμπυλών δεν είναι τόσο διαδεδομένη όσο αυτή των Ελλειπτικών Καμπυλών, επομένως, δεν υπάρχουν αντίστοιχες βάσεις δεδομένων που να έχουν ορίσει προσεκτικά τις παραμέτρους των Καμπυλών που πρέπει να επιλέγονται. Για αυτό τον λόγο δεν είναι τόσο εύκολο να χρησιμοποιηθούν στη βιομηχανία ακόμα, αφού βρίσκονται σε ερευνητικό στάδιο. Επίσης, οι αλγόριθμοι που υπάρχουν για τον υπολογισμό της τάξης της Ομάδας που προκύπτει από την Κλάση των Reduced Divisor δεν είναι ιδιαίτερα αποδοτικοί και αυτό δυσκολεύει τη διαδικασία παραγωγής ασφαλών Υπερελλειπτικών Καμπυλών.

Πάραυτα, το μικρό μέγεθος κλειδιών που προκύπτουν, αλλά και οι βελτιώσεις στην αριθμητική τους τις καθιστούν ιδιαίτερα ενδιαφέρουσες για συστήματα περιορισμένων πόρων, όπως είναι τα ενσωματωμένα συστήματα και τα Αυτό-Οργανούμενα Δίκτυα Οχημάτων (VANETs), γι' αυτό και τα τελευταία χρόνια ο συγκεκριμένος κλάδος έχει αρχίσει ξανά να τραβάει τα βλέμματα των μηχανικών.

2.3 Πρωτόκολλα ασύρματης επικοινωνίας

Σε αυτή την ενότητα παρουσιάζονται τα πρωτόκολλα επικοινωνίας WiFi και WAVE, το οποίο είναι βασισμένο στο WiFi, για την καλύτερη κατανόηση της διαδικασίας επικοινωνίας των οχημάτων στα Αυτό-Οργανούμενα Δίκτυα Οχημάτων (VANETs).

2.3.1 WiFi

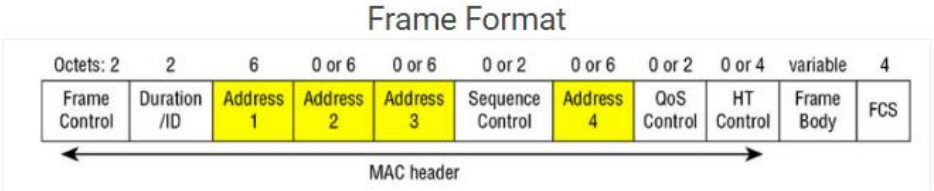
Το πιο διαδεδομένο πρωτόκολλο ασύρματης επικοινωνίας είναι το WiFi, ή αλλιώς ασύρματο LAN, το οποίο δίνει τη δυνατότητα για σύνδεση συσκευών σε τοπικό δίκτυο, αλλά και με τη μορφή gateway για σύνδεση στο διαδίκτυο. Το WiFi ορίζεται από τα πρότυπα 802.11 από την IEEE και η διάδοση του έφερε επανάσταση στις ασύρματες επικοινωνίες.

Το πρωτόκολλο υποστηρίζει τα εύρη ζώνης των 2.4 GHz και των 5 GHz στη γενική περίπτωση, ωστόσο το πρότυπο 802.11p που δημιουργήθηκε για τις ανάγκες του πρωτοκόλλου WAVE λειτουργεί σε εύρος ζώνης 5.9 GHz. Το WiFi λειτουργεί στο φυσικό επίπεδο και στο επίπεδο Ζεύξης δεδομένων. Στο φυσικό επίπεδο, το WiFi χρησιμοποιεί την τεχνική της Ορθογώνιας Πολύπλεξης Διαίρεσης Συχνοτήτων, ή αλλιώς OFDM για την εκπομπή του σήματος στον αέρα. Επιπλέον, εφαρμόζουν τη μέθοδο CSMA/CA (Carrier-Sense Multiple Access with Collision Avoidance), η οποία

υλοποιεί έλεγχο διαθεσιμότητας του μέσου εκπομπής για τη μείωση των συγκρούσεων και σε περίπτωση μεγάλης κίνησης χρησιμοποιεί την εκθετική οπισθοχώρηση (exponential backoff) για την αποσυμφόρηση του μέσου [13].

Το πλαίσιο, στη βασική του μορφή, που χρησιμοποιείται στο WiFi αποτελείται από τα εξής πεδία [14]:

1. Έλεγχος πλαισίου. Το πεδίο ελέγχου πλαισίου έχει μέγεθος 2 bytes και χωρίζεται σε υπό-πεδία των 1, 2 και 4 bits.
2. Διάρκεια/ID: Το πεδίο περιέχει την διάρκεια του επόμενου πλαισίου προς μετάδοση.
3. Διεύθυνση 1: Είναι η διεύθυνση MAC προορισμού.
4. Διεύθυνση 2: Είναι η διεύθυνση MAC της συσκευής που μεταδίδει το πλαίσιο.
5. Διεύθυνση 3: Είναι η διεύθυνση MAC του δρομολογητή.
6. Αριθμός Ακολουθίας: Το πεδίο χρησιμεύει για να αναδεικνύει τη σειρά των πακέτων σε περίπτωση κατάτμησης και για τη χρήση και την αναγνώριση αναμετάδοσης πλαισίων.
7. Διεύθυνση 4: Χρησιμοποιείται στη λειτουργία ad hoc.
8. Έλεγχος Ποιότητας Υπηρεσίας (QoS): Πεδίο που χρησιμοποιείται για την εξασφάλιση της βελτιστοποίησης υπηρεσίας ανάλογα με το επιθυμητό επίπεδο, που εξαρτάται από το είδος της επικοινωνίας που εξελίσσεται. Και αυτό χωρίζεται σε υπό-πεδία.
9. HT έλεγχος: Περιέχει τις παραμέτρους που είναι αναγκαίες για την μετάδοση των πλαισίων στο μέσο και επιτρέπει χρήση εξελιγμένων τεχνικών για να βελτιώσει την επίδοση της μετάδοσης [15].
10. Ωφέλιμο φορτίου: Συνήθως αποτελείται από ένα πακέτο IP ή ένα πακέτο ARP. Το μέγεθός του μπορεί να φτάσει ως 2312 bytes αλλά συνήθως περιορίζεται σε μεγέθη μικρότερα των 1500 bytes.
11. Frame check Sequence ή FCS: Μηχανισμός ελέγχου λαθών του πλαισίου.



Εικόνα 8: Το γενικό πλαίσιο του WiFi [14]

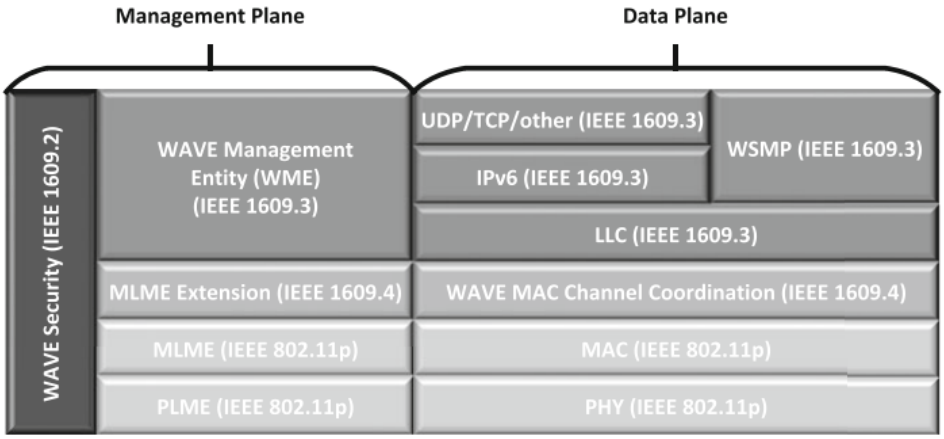
2.3.2 WAVE

Με την εξέλιξη των Δικτύων Οχημάτων τα τελευταία χρόνια, δημιουργήθηκε η ανάγκη για τη δημιουργία ενός συστήματος που θα καλύπτει τις ανάγκες τους. Γι' αυτό δημιουργήθηκε το συστήματος ασύρματης σύνδεσης σε περιβάλλοντα οχημάτων, το λεγόμενο WAVE (wireless access in vehicular environments) [16]. Το WAVE λειτουργεί στα 5.9 GHz, αφού αποτελεί έναν συνδυασμό των προτύπων IEEE

1609 και το προτύπου IEEE 802.11p που αναφέρθηκε προηγουμένως. Το WAVE σύστημα, δηλαδή, βασίζεται στο WiFi στο φυσικό στρώμα και επεκτείνεται για να καλύπτει όλες τις ανάγκες των δικτύων οχημάτων. Το WAVE έχει τη δυνατότητα εμβέλειας έως και 500 μέτρα.

Το εύρος ζώνης που αξιοποιεί είναι μεγέθους 75 MHz και μοιράζεται σε 7 κανάλια των 10 MHz. Κάθε κανάλι αξιοποιείται με διαφορετικό τρόπο. Για παράδειγμα, το κανάλι 172 είναι αφιερωμένο στην υλοποίηση επικοινωνίας οχήματος με οχήματα για την ασφάλεια της μετακίνησης και την αποφυγή ατυχημάτων. Το κανάλι 178 χρησιμοποιείται αποκλειστικά για μηνύματα WSMP (WAVE Short Message Protocol), τα οποία δίνουν τη δυνατότητα για πολύ γρήγορη ανταλλαγή μηνυμάτων.

Η στοίβα πρωτοκόλλων του WAVE ορίζεται ολοκληρωτικά από τα πρότυπα IEEE 1609 και IEEE Std 802.11p και χωρίζεται σε δύο επίπεδα. Το ένα επίπεδο αφορά τα δεδομένα με τα πρωτόκολλο των ανώτερων επιπέδων και το δεύτερο επίπεδο τις συναρτήσεις διαχείρισης που ορίζονται στο WAVE.



Εικόνα 9: Η στοίβα πρωτοκόλλων του WAVE [16]

Το πρότυπο IEEE Std 1609.3 ορίζει δύο στοίβες πρωτοκόλλων στο επίπεδο των δεδομένων, το ένα αφορά το IPv6 πρωτόκολλο και το άλλο το WSMP που προαναφέρθηκε, τα οποία είναι και τα δύο βασισμένα στα ίδια πρωτόκολλα στα χαμηλότερα επίπεδα, δηλαδή στο φυσικό, στο επίπεδο MAC και στο επίπεδο ζεύξης. Το επίπεδο της διαχείρισης με τη θέσπιση του MLME (MAC layer management entity) κατανέμει τα πακέτα δεδομένων σε κατηγορίες/ουρές με βάση το είδος της πρόσβασης για την διαχείριση της Ποιότητας Υπηρεσίας (QoS). Το φυσικό στρώμα του WAVE ακολουθεί την OFDM τεχνική που χρησιμοποιεί και το γενικό WiFi για την αναμετάδοση. Το άνω στρώμα MAC που βασίζεται στο πρότυπο IEEE 1609.4, διαχειρίζεται τη χρήση των καναλιών και για να ορίζει στρατηγικές πολλαπλών καναλιών κατά τη μετάδοση δεδομένων που είτε αφορούν, είτε όχι την ασφάλεια.

Το WAVE επιτυγχάνει, τελικά, να έχει γρήγορη παράδοση, μιας και εφόσον τα συγκεκριμένα δίκτυα έχουν την ιδιαιτερότητα να εξυπηρετούν χρήστες/οχήματα που έχουν μεγάλες ταχύτητες και μπορεί να αλλάζουν γρήγορα σημεία πρόσβασης.

Επίσης, αφομοιώνει την τεχνολογία του GeoNetworking και επιτρέπει στα οχήματα να επικοινωνούν με ευκολία τη γεωγραφική τους τοποθεσία. Τα πρότυπα IEEE 1609, επίσης, θεσπίζουν μηχανισμούς για την ενίσχυση της ιδιωτικότητας κατά την επικοινωνία, πράγμα που είναι ιδιαίτερα επιθυμητό για τους λόγους που αναλύθηκαν παραπάνω στην εργασία. Επίσης, το WAVE αφομοιώνει τις τεχνικές της Ποιότητας Υπηρεσίας (QoS), για να διαχωρίσει την κίνηση σε επίπεδα προτεραιότητας ανάλογα με τη φύση της εφαρμογής. Αυτό δίνει τη δυνατότητα για την υλοποίηση εφαρμογών πραγματικού χρόνου που ενδέχεται να μοιράζονται μεγάλα αρχεία βίντεο και εικόνων.

2.4 Ασφάλεια και ιδιωτικότητα

Παραπάνω, αναλύθηκαν οι πιο βασικές κρυπτογραφικές έννοιες, καθώς και οι βασικές αρχές της κρυπτογραφίας των Υπερελλειπτικών Καμπυλών. Ωστόσο, για την καλύτερη αξιοποίησή τους, είναι αναγκαίο να μελετηθούν οι κίνδυνοι που εγείρονται κυρίως στα Δίκτυα Οχημάτων, ώστε να προσαρμοστούν οι τεχνικές στις ανάγκες τους και να τα ασφαλίσουν με επιτυχία. Επομένως, παρουσιάζονται οι τύποι των επιτιθέμενων που επιθυμούν να αποκτήσουν πρόσβαση στα δεδομένα των χρηστών, καθώς και κάποιες από τις γνωστές τεχνικές τους.

2.4.1 Τύποι επιτιθέμενων

Οι κυριότεροι τύποι επιτιθέμενων στα Δίκτυα Οχημάτων, αλλά και γενικά, είναι οι εξής:

- Ενεργητικός/Παθητικός (Active/Passive): Ο ενεργός επιτιθέμενος προκειμένου να παραπλανήσει τους χρήστες του δικτύου και να κλέψει τα δεδομένα τους ονομάζεται ενεργητικός. Ο ενεργός επιτιθέμενος έχει πρόσβαση στο εσωτερικό τους δικτύου. Από την άλλη, ο παθητικός επιτιθέμενος δεν αποσκοπεί να επικοινωνήσει με τους χρήστες του δικτύου, ούτε να κάνει φανερή την παρουσία του στους υπόλοιπους. Προσπαθεί από το παρασκήνιο να παραβιάσει την επικοινωνία των υπόλοιπων χρηστών κρυφακούγοντάς την [17] [18].
- Εξωτερικός/Εσωτερικός (Outsider/Insider): Ένας εξωτερικός επιτιθέμενος δεν βρίσκεται εντός του Δικτύου Οχημάτων και δεν θεωρείται ως έγκυρος χρήστης του. Επομένως, οι δυνατότητές του είναι περιορισμένες, αφού το μόνο που μπορεί να κάνει είναι να κρυφακούει την επικοινωνία των υπολοίπων ή να προσπαθήσει να αποτρέψει την επικοινωνία δημιουργώντας αυξημένη κίνηση στο δίκτυο. Από την άλλη, ένας εσωτερικός χρήστης θεωρείται ως ένας επικυρωμένος χρήστης του δικτύου που, όμως, έχει βλέψεις να κλέψει τα δεδομένα των υπόλοιπων χρηστών. Ένας επιτιθέμενος τέτοιου τύπου έχει τη δυνατότητα να εκτελέσει πολλές περισσότερες επιθέσεις και να παρέμβει με περισσότερους τρόπους στην

επικοινωνία, μέχρι και να παραπλανήσει τους υπόλοιπους χρήστες. Στα Δίκτυα Οχημάτων, συγκεκριμένα, υπάρχει μία κατηγορία τέτοιων χρηστών που έχουν τη δυνατότητα να επέμβουν στο λογισμικό του κώδικα, εισάγοντας κακόβουλο λογισμικό στα οχήματα [17] [18].

- Στατικός/Δυναμικός (Static/Adaptive): Ένας στατικός επιτιθέμενος απλώς επιλέγει τον τρόπο επίθεσής του και ακολουθεί αυτόν τον τρόπο ανεξαρτήτως της αποτελεσματικότητάς του. Αντίθετα, ο δυναμικός επιτιθέμενος παρατηρεί του συμπεριφορά του Δικτύου κατά την επίθεσή του και επιλέγει να προσαρμοστεί σε αυτό, αλλάζοντας τις μεθόδους επίθεσής του με σκοπό να γίνει πιο αποτελεσματικός [17] [18].
- Μοχθηρός/Λογικός (Malicious/Rational): Ένας μοχθηρός επιτιθέμενος δεν έχει κάποιο όφελος από την επίθεση που πραγματοποιεί και δεν αποσκοπεί να κερδίσει κάτι από αυτή. Επομένως, τον ενδιαφέρει απλώς να προκαλέσει προβλήματα στη λειτουργία του δικτύου. Αντίθετα, ένας λογικός επιτιθέμενος προσπαθεί να κλέψει δεδομένα για δικό του όφελος και επομένως προτιμά να δρα στο παρασκήνιο και με «εξυπνάδα», ώστε να μην γίνει αντιληπτός και να είναι πιο αποτελεσματικός [17] [18].
- Καθολικός/Τοπικός (Global/Local): Ένας τοπικός επιτιθέμενος έχει πρόσβαση μόνο σε έναν περιορισμένο αριθμό πόρων του δικτύου και δεν έχει τη δυνατότητα να επεκταθεί περαιτέρω. Ένας καθολικός, από την άλλη, έχει πρόσβαση στο σύνολο του δικτύου και μπορεί να το επηρεάσει καθολικά [17] [18].

2.4.2 Μέθοδοι επιθέσεων

Οι επιτιθέμενοι έχουν στη διάθεσή τους πληθώρα από τρόπους επίθεσης στα Δίκτυα Οχημάτων. Κάποιοι από αυτούς είναι οι εξής:

- Επιθέσεις στην Εμπιστευτικότητα (Attacks on Confidentiality): Οι επιθέσεις στην εμπιστευτικότητα στοχεύουν στο να κρυφακούσουν στην επικοινωνία που λαμβάνει χώρα στο Δίκτυο με σκοπό να κλέψουν ευαίσθητες πληροφορίες, όπως είναι οι κωδικοί. Ο επιτιθέμενος μπορεί να υποδυθεί έναν κόμβο του Δικτύου ή μία RSU [18].
- Επιθέσεις στη διαθεσιμότητα (Availability attacks): Οι επιθέσεις στη διαθεσιμότητα αποσκοπούν να αποτρέψουν τη δυνατότητα επικοινωνίας των χρηστών με διάφορους τρόπους. Ο πιο διαδεδομένος τρόπος είναι η επίθεση Άρνησης Υπηρεσίας (DoS), στην οποία ο επιτιθέμενος κατακλύζει το δίκτυο με άχρηστα μηνύματα με σκοπό την πλήρη κατανάλωση των πόρων του και την αδυναμία εξυπηρέτησης άλλων χρηστών. Αν η επίθεση εκτελείται από πολλούς κόμβους ταυτόχρονα η επίθεση ονομάζεται Κατανεμημένη Άρνηση Υπηρεσίας (DDoS) [18].
- Επιθέσεις στη δρομολόγηση (Routing attacks): Οι επιθέσεις εκμεταλλεύονται την εγγενή πολυπλοκότητα των μηχανισμών δρομολόγησης στα Δίκτυα

Οχημάτων. Ο επιτιθέμενος μπορεί να προσελκύσει θύματα παρουσιάζοντας τον εαυτό του ως τον «κοντινότερο» κόμβο στον κόμβο προορισμού (συνήθως η RSU) (Black Hole attack). Όταν δύο ή περισσότεροι επιτιθέμενοι ακολουθούν την παραπάνω τακτική και βρίσκονται κοντά μεταξύ τους μπορούν να ενισχύσουν περαιτέρω τη δράση τους διαφημίζοντας το μονοπάτι τους ως το συντομότερο (Worm Hole attack) [18].

- Επιθέσεις στην αυθεντικότητα των δεδομένων (Data authenticity attacks): Σε αυτή την μέθοδο ο επιτιθέμενος επικεντρώνεται στην εκμετάλλευση του περιεχομένου του πακέτου. Ο επιτιθέμενος μπορεί να αποστείλει πολλές φορές το ίδιο μήνυμα δηλητηριάζοντας έτσι τους πίνακες θέσεις των κόμβων (Replay attack), μπορεί να στείλει ψευδείς πληροφορίες για την θέση του δημιουργώντας σύγχυση στο δίκτυο (Position Faking – Sybil Attack) ή μπορεί να μεταβάλλει το περιεχόμενο μηνυμάτων που στέλνονται με σκοπό να μεταδώσει ψευδείς πληροφορίες (Message Tampering) [18].

Εφόσον στην παρούσα εργασία οι κρυπτογραφικές τεχνικές που χρησιμοποιούνται αφορούν τις Ελλειπτικές/Υπερελλειπτικές Καμπύλες είναι σημαντικό να παρουσιαστούν κάποιες γνωστές επιθέσεις σε αυτές [19]. Συγκεκριμένα:

- Επιθέσεις Πλευρικού Καναλιού (Side-Channel Attacks): Οι επιθέσεις πλευρικού καναλιού δεν αξιοποιούν άμεσα κάποια αδυναμία του συστήματος, αλλά κάποιο αποτύπωμα που αφήνει ένας αλγόριθμος στο λογισμικό ή στην ενέργεια που καταναλώνεται. Για παράδειγμα, στα κρυπτογραφικά συστήματα Ελλειπτικών Καμπυλών, όπως παρουσιάστηκε η πράξη που ορίζει την Ομάδα είναι η «πρόσθεση» δύο σημείων της καμπύλης με την έννοια που περιγράφηκε στο κεφάλαιο 2.2.3. Ο διπλασιασμός ενός σημείου, ωστόσο, στην πράξη υπολογίζεται με διαφορετικό τρόπο από τη γενική περίπτωση της πρόσθεσης και είναι πιο γρήγορος. Με αυτόν τον τρόπο, κάποιος επιτιθέμενος που θα αξιοποιούσε το χρονικό αποτύπωμα που αφήνει ο βαθμωτός πολλαπλασιασμός, θα μπορούσε να λάβει πληροφορία που θα τον βοηθήσει να λύσει το Πρόβλημα του Διακριτού Λογάριθμου πιο γρήγορα.
- Επίθεση άκυρης καμπύλης (Invalid curve attack): Σε αυτή την επίθεση, ένας κακόβουλος χρήστης παρεμβάλλεται στην επικοινωνία και εξαναγκάζει τους χρήστες του Δικτύου να χρησιμοποιήσουν καμπύλες, οι οποίες δεν είναι αρκετά ασφαλείς με αποτέλεσμα να κάνουν πιο εύκολη τη διαδικασία της λύσης του Διακριτού Λογάριθμου. Αντίστοιχες είναι και οι επιθέσεις άκυρου σημείου (Invalid Point Attacks), όπου σαν γεννήτρια επιλέγεται ένα σημείο που δημιουργεί μία Ομάδα μικρής τάξης.
- Επίθεση ανεστραμμένης καμπύλης (Twisted curve attack): Έχει παρατηρηθεί πως πολλές από τις καμπύλες που θεωρούνται ασφαλείς, έχουν κρυπτογραφικά αδύναμη ανάστροφη καμπύλη. Αυτό μπορεί να οδηγήσει σε διαρροή πληροφορίας για το ιδιωτικό κλειδί.

Οι παραπάνω είναι μόνο κάποιες από τις γνωστές επιθέσεις σε Δίκτυα Οχημάτων (VANETs) και στην κρυπτογραφία των Ελλειπτικών/Υπερελλειπτικών Καμπυλών. Επομένως, είναι ιδιαίτερα σημαντικό να λαμβάνονται υπόψη κατά την οργάνωση ενός δικτύου αλλά και κατά την επιλογή των παραμέτρων των καμπυλών που το ασφαλίζουν. Στην παρούσα εργασία δίνεται έμφαση επιθέσεις στην εμπιστευτικότητα, στη διαθεσιμότητα, στην αυθεντικότητα των δεδομένων και στην ασφάλεια των καμπυλών. Περαιτέρω εμβάθυνση στο θέμα προτείνεται με μελέτη της αντίστοιχης βιβλιογραφίας.

Κεφάλαιο 3: Συναφής Βιβλιογραφία

Σε αυτό το κεφάλαιο γίνεται η παρουσίαση και ανάλυση με συνοπτικό τρόπο των δημοσιεύσεων που χρησιμοποιήθηκαν για την υλοποίηση της διπλωματικής εργασίας. Αρχικά, γίνεται η παρουσίαση των σχημάτων εξασφάλισης ιδιωτικότητας σε VANETs που μελετήθηκαν και η σύγκρισή τους. Στη συνέχεια, θα παρουσιαστεί η βιβλιογραφία που αφορά τις μεθόδους κρυπτογράφησης με υπερελλειπτικές καμπύλες.

3.1 Ασφαλή και αποδοτικά σχήματα για ανταλλαγή μηνυμάτων σε VANETs

3.1.1 Σχήματα αυθεντικοποίησης

Με τη συνεχή εξέλιξη του IoT και συνεπώς του IoV πολλά σχήματα έχουν προταθεί για την εξασφάλιση της ασφαλούς και της αποδοτικής επικοινωνίας σε VANETs. Μερικά από αυτά βασίζονται αποκλειστικά στην αυθεντικοποίηση των χρηστών, άλλα αντιμετωπίζουν και την επεξεργασία και την μεταβίβαση των μηνυμάτων, ενώ άλλα προσπαθούν να βελτιώσουν την επίδοση της διαδικασίας αυθεντικοποίησης των χρηστών και κατανομής των μηνυμάτων στο δίκτυο. Για αρχή παρουσιάζονται τα σχήματα που αφορούν την αυθεντικοποίηση.

Το σχήμα EAAP: Efficient Anonymous Authentication With Conditional Privacy-Preserving [20] για VANETs προτάθηκε από τους Maria Azees, Pandi Vijayakumar και Lazarus Deborah τον Σεπτέμβριο του 2017 και αποσκοπεί στην υλοποίηση ενός αποδοτικού σχήματος ανώνυμης αυθεντικοποίησης οχημάτων και RSU σε VANETs. Επιπλέον, δίνουν τη δυνατότητα για τον εντοπισμό κακόβουλων χρηστών και τον αποκλεισμό τους από το δίκτυο. Η διαδικασία αυθεντικοποίησης είναι βασισμένη στην κρυπτογραφική τεχνική του bilinear-pairing που αποτελεί μία από τις πιο διαδεδομένες μεθόδους κατανομής δημοσίων κλειδιών.

Η διαδικασία αυθεντικοποίησης γίνεται με διαφορετικό τρόπο για τα οχήματα και τις RSU. Αρχικά, η Έμπιστη Αρχή (αναφέρεται ως TA – Trusted Authority, όρος αντίστοιχος της CA) γνωστοποιεί σε όλους τις παραμέτρους που θα χρησιμοποιηθούν για τη δημιουργία των πιστοποιητικών και των υπογραφών. Ένα όχημα, για να μπορέσει να αυθεντικοποιηθεί αρχικά πρέπει να εγγραφεί στην TA και να λάβει το «ψεύτικο» αναγνωριστικό του, DID (Dummy ID). Ωστόσο, η TA κρατά πληροφορία για την σύνδεση του «ψεύτικου» αναγνωριστικού με την πραγματική ταυτότητα του οχήματος, ώστε σε περίπτωση κακόβουλης συμπεριφοράς, να μπορέσει να το αποκλείσει από το δίκτυο. Όταν ένα όχημα εισέλθει δημιουργεί το ανώνυμο πιστοποιητικό του, το υπογράφει και το μοιράζεται μαζί με το δημόσιο κλειδί του και διάφορες άλλες αναγκαίες παραμέτρους για την επικύρωση.

Τα ανώνυμα πιστοποιητικά μίας RSU παράγονται άμεσα από την TA και γνωστοποιούνται από αυτή στην RSU. Προηγουμένως, η RSU πρέπει και αυτή να έχει εγγραφεί στο σύστημα της TA για να λάβει τα αναγνωριστικά της και να είναι δυνατός ο εντοπισμός της σε περίπτωση κακόβουλης συμπεριφοράς. Η RSU υπογράφει το πιστοποιητικό της με ένα προσωρινό μυστικό κλειδί και γνωστοποιεί το πιστοποιητικό της, το προσωρινό δημόσιο κλειδί της και την υπογραφεί σε όλα τα οχήματα στο δίκτυο.

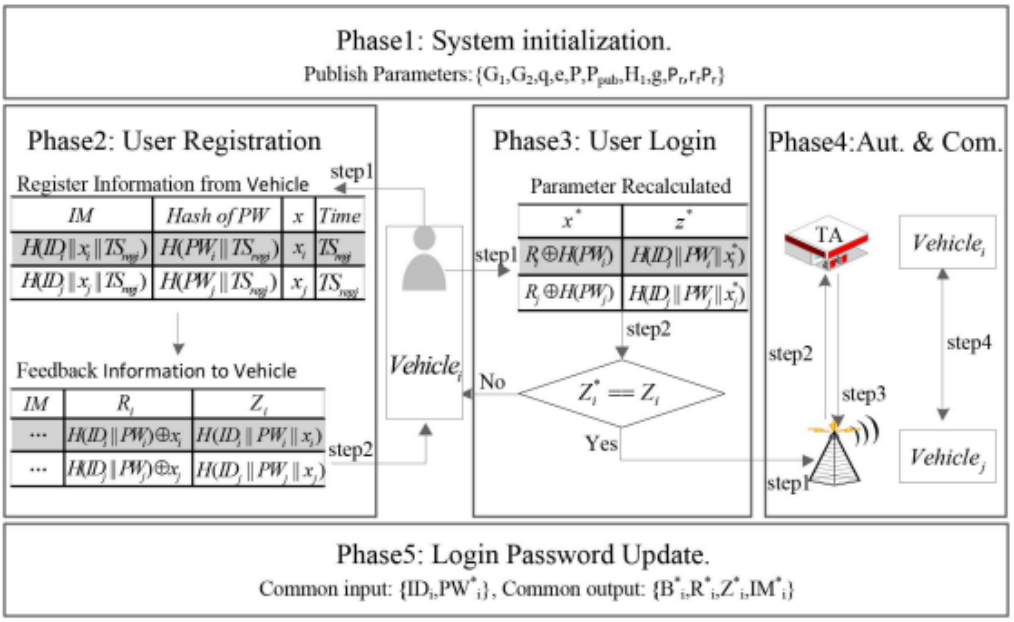
Η διαδικασία δημιουργίας και επικύρωσης του ανώνυμου πιστοποιητικού και της υπογραφής έχει οριστεί με τέτοιον τρόπο, ώστε να ελαχιστοποιεί τις απαιτούμενες πράξεις και κατάφερε να επιτύχει χρόνους σημαντικά καλύτερους από τα υπόλοιπα σχήματα αυθεντικοποίησης της εποχής του. Είναι ασφαλές ενάντια σε επιθέσεις αυθεντικοποίησης, όπως η επίθεση Impersonation, σε επιθέσεις στην αυθεντικότητα των δεδομένων, όπως η επίθεση bogus message και τέλος εξασφαλίζει την ιδιωτικότητα θέσης [18].

Το ίδιο έτος, οι Yanbing Liu κ.α. παρουσίασαν το δικό τους σχήμα δυαδικής αυθεντικοποίησης, το PPDAS [21]. Το σχήμα αξιοποιεί τις μεθόδους κρυπτογράφησης με βάση το αναγνωριστικό του οχήματος καθώς και κρυπτογράφηση δημοσίου κλειδιού. Αυτό που εισάγει, ωστόσο, είναι η φήμη του οχήματος (vehicle reputation). Επίσης, για να ξεκινήσει η επικοινωνία μεταξύ δύο οχημάτων, η αυθεντικοποίηση περνάει από την TA, η οποία παράγει τα απαιτούμενα κλειδιά και υπογραφές για την αρχικοποίηση της επικοινωνίας.. Αφού, τα οχήματα έχουν εγγραφεί στην Έμπιστη Αρχή και έχουν λάβει τις παραμέτρους τους, πραγματοποιούν σύνδεση με αυτήν μέσω της RSU (login). Αν αυτό γίνει με επιτυχία, τα οχήματα δημιουργούν τα ψεύδο-αναγνωριστικά τους, από τα οποία μόνο η RSU και η TA έχουν τη δυνατότητα να υπολογίσουν την πραγματική ταυτότητά τους με τη χρήση των ιδιοτήτων του bilinear pairing. Παράλληλα, χρονικές στάμπες χρησιμοποιούνται στα μηνύματα για την αποφυγή των επιθέσεων επανάληψης (Replay Attack), καθώς και η τεχνική του MAC (Message Authentication Code) για την αποφυγή των επιθέσεων αλλοίωσης μηνύματος (Message Tampering)

Η αυθεντικοποίηση, εδώ, πραγματοποιείται σε δύο στάδια Αρχικά, τα οχήματα στέλνουν, μέσω της RSU, η οποία πραγματοποιεί κάποιους αρχικούς υπολογισμούς για την αυθεντικοποίηση, στην TA τα ψεύδο-αναγνωριστικά τους, μαζί με τα δημόσια κλειδιά τους, τη χρονική στάμπα κ.α. Αφού επικυρωθεί η χρονική στάμπα από την TA, τότε υπολογίζει τον κώδικα MAC με βάση τα πραγματικά αναγνωριστικά των οχημάτων και τον συγκρίνει με τον κώδικα MAC που έχει λάβει. Αν η σύγκριση επιτύχει, τότε τα οχήματα επικυρώνεται σε πρώτο στάδιο. Στη συνέχεια, ελέγχεται η φήμη των οχημάτων. Η αξιολόγηση των οχημάτων γίνεται με τεχνικές που χρησιμοποιούν πίνακες εμπιστοσύνης και ανάλογα το ζευγάρι των οχημάτων που επικοινωνούν προκύπτει ένα μέσο επίπεδο εμπιστοσύνης. Επομένως, η αυθεντικοποίηση σε δεύτερο στάδιο γίνεται μέσω του επιπέδου εμπιστοσύνης των οχημάτων. Αφού τελειώσει η επικοινωνία τους, τα οχήματα αξιολογούν το ένα το

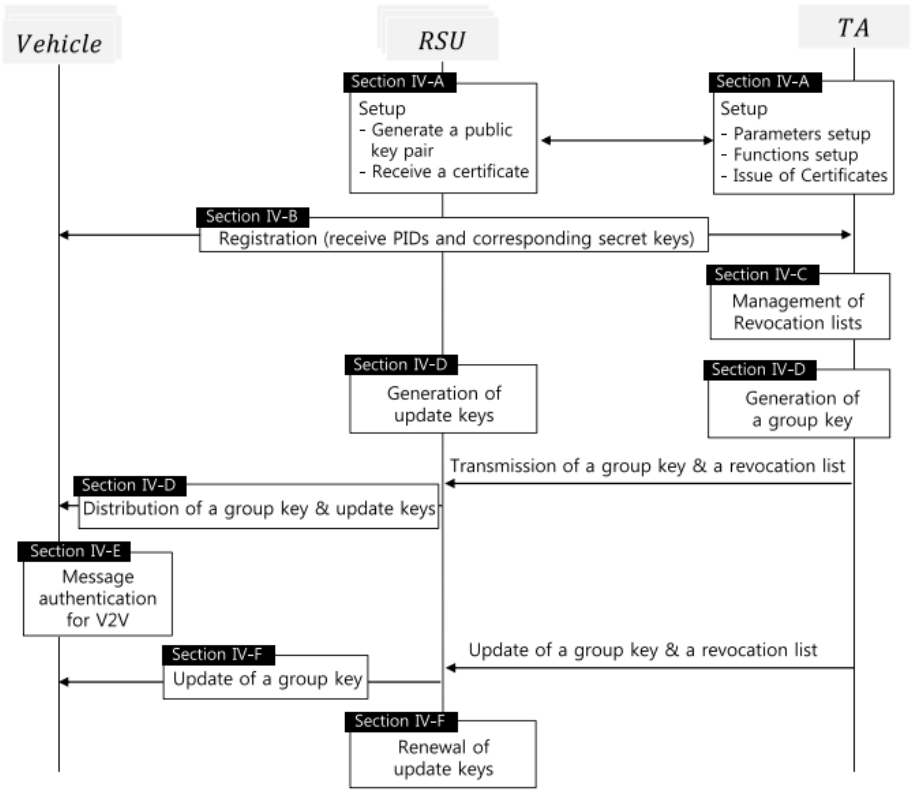
άλλο και ενημερώνουν την TA, μέσω της RSU, για την ποιότητά της. Έτσι, η TA ανανεώνει τους πίνακες εμπιστοσύνης.

Το συγκεκριμένο σχήμα ενισχύει την ασφάλεια της αυθεντικοποίησης χρησιμοποιώντας την αυθεντικοποίηση σε 2 στάδια. Είναι ασφαλές από επιθέσεις Man-in-the-Middle και από επιθέσεις αλλοίωσης μηνύματος και επανάληψης μηνύματος. Επίσης, διατηρεί την ανωνυμία των οχημάτων με τη χρήση των ψευδο-αναγνωριστικών, εφόσον για να γίνει η επικοινωνία ενός οχήματος με ένα άλλο δεν είναι απαραίτητο να γνωρίζουν την πραγματική ταυτότητά τους. Ωστόσο, το σχήμα δεν είναι ιδιαίτερα αποδοτικό, αφού η διαδικασία αυθεντικοποίησης απαιτεί πολλές και χρονοβόρες πράξεις, καθώς και πολλά μηνύματα μεγάλου μεγέθους που πρέπει πάντοτε να επικυρώνονται από την TA.



Εικόνα 10: Το σχήμα αυθεντικοποίησης PPDAS [21]

Το 2018 οι Hyo Jin Jo κ.α. [22] πρότειναν ένα αντίστοιχο σχήμα αυθεντικοποίησης, το οποίο αξιοποιεί τη συνεργασία μεταξύ των οχημάτων και της RSU για την αυθεντικοποίηση των μηνυμάτων και των χρηστών. Η μετάδοση και διαχείριση των κλειδιών γίνεται με τη χρήση δυαδικών δέντρων κλειδιών. Για την αυθεντικοποίηση των μηνυμάτων όλα τα οχήματα που λαμβάνουν ένα μήνυμα από κάποιο κοντινό όχημα ανά κάποιο χρονικό διάστημα επικυρώνουν την αυθεντικότητά τους και μετά δημοσιεύουν τα αποτελέσματά τους στα κοντινά τους οχήματα. Στη συνέχεια όλα τα οχήματα επικυρώνουν τα αποτελέσματα των διπλανών τους και έτσι τελικά λαμβάνουν μία συνολική συνεργατική πληροφορία για τα επικυρωμένα μηνύματα στο δίκτυο.



Εικόνα 11: Συνεργατική Αυθεντικοποίηση [22]

Η RSU, από την άλλη, είναι υπεύθυνη για την ανάκληση των κακόβουλων ή άκυρων χρηστών, πράγμα που επιτυγχάνεται με τη χρήση ομαδικών κλειδιών τα οποία η RSU γνωστοποιεί μόνο σε έγκυρους χρήστες. Αν κάποιος χρήστης έχει ανακληθεί, τότε η υπογραφή του μηνύματος του και συμπερασματικά η επικύρωσή του από τους υπόλοιπους χρήστες αποτυγχάνει.

Το παραπάνω σχήμα βελτιώνει τη μεταφορά της Λίστας Ανάκλησης, η οποία μπορεί να λάβει μεγάλες διαστάσεις στο Δίκτυο Οχημάτων, προσθέτει έναν αποδοτικό τρόπο ανάκλησης, ανανέωσης και διαμοιρασμού δημοσίων κλειδιών και εντοπισμού κακόβουλων χρηστών στο δίκτυο. Όπως και το EAAP, επιτυγχάνει την αυθεντικοποίηση και την ακεραιότητα των μηνυμάτων, την ανωνυμία και την ιδιωτικότητα των χρηστών και λύνει αποδοτικά το πρόβλημα της ανάκλησης [18].

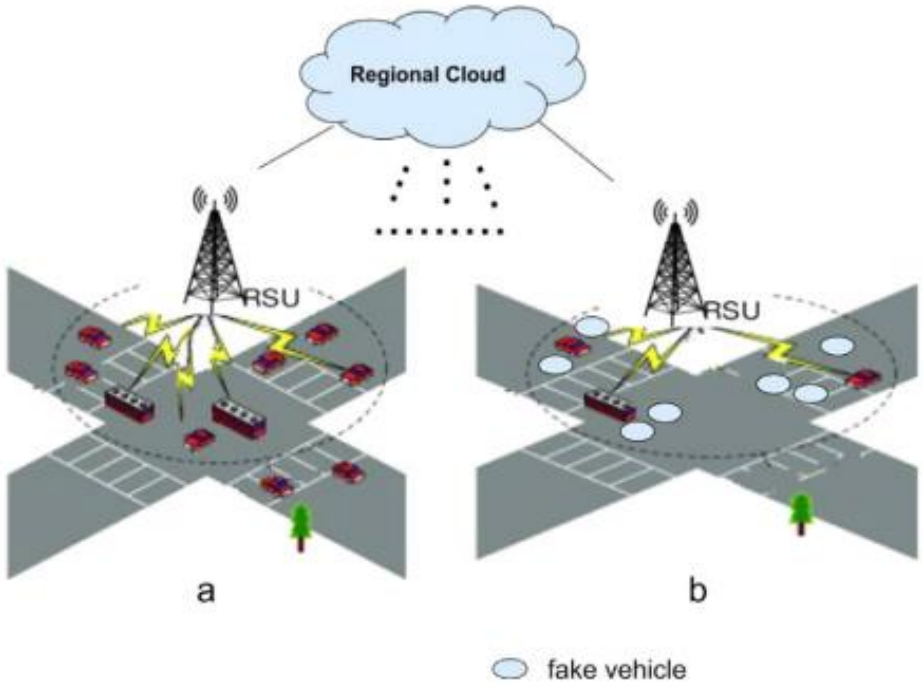
Το βασικό μειονέκτημα των παραπάνω σχημάτων είναι πως για την αυθεντικοποίηση απαιτούν την πραγματοποίηση επικοινωνίας των οχημάτων μεταξύ τους, με την RSU και στην περίπτωση του PPDAS, με την Έμπιστη Αρχή (TA – CA). Αυτό δημιουργεί έντονη δικτυακή κίνηση και κατακλύζει το δίκτυο με πακέτα, ενώ επίσης προσθέτει χρονική καθυστέρηση στην επικοινωνία, η οποία είναι επιθυμητό να γίνεται με ταχύτητα.

3.1.2 Σχήματα ασφαλής διάδοσης μηνυμάτων

Το 2021, οι Hassan Mistareehi κ.α. [23] πρότειναν μία αρχιτεκτονική για ασφαλή διάδοση μηνυμάτων σε δίκτυα οχημάτων στο σύννεφο (vehicular cloud). Η αρχιτεκτονική αποσκοπεί κυρίως στη βελτίωση της απόδοσης της αυθεντικοποίησης και της διάδοσης των μηνυμάτων σε περίπτωση που η πυκνότητα των οχημάτων σε μία περιοχή γίνεται μεγάλη. Το συγκεκριμένο σχήμα αξιοποιεί τόσο συμμετρική και ασύμμετρη κρυπτογράφηση για την μεταφορά κλειδιών και μηνυμάτων. Επίσης, σε περίπτωση που ένα όχημα δεν βρίσκεται στην εμβέλεια μίας RSU, τότε το μήνυμα/αίτημά του μεταβιβάζεται στην RSU μέσω άλλων, κοντινών οχημάτων με τη χρήση ενός αλγόριθμου δρομολόγησης, συγκεκριμένα τον αλγόριθμο AODV.

Όταν ένα αυτοκίνητο εγγράφεται σε μία ένα τοπικό σύννεφο (regional cloud – RC) λαμβάνει ένα ζεύγος κλειδιών, καθώς και τα δημόσια κλειδιά των RSUs της περιοχής. Όλα τα μηνύματα στο σχήμα αυθεντικοποιούνται με τη χρήση ψηφιακής υπογραφής, που υπογράφεται με το ιδιωτικό κλειδί και επικυρώνεται με το δημόσιο. Η κρυπτογράφηση των ευαίσθητων μηνυμάτων γίνεται με τη χρήση συμμετρικού αλγόριθμου κρυπτογράφησης, όπου το κλειδί παράγεται τη στιγμή της αποστολής και κρυπτογραφείται με ασύμμετρο τρόπο με το δημόσιο κλειδί της RSU. Όταν ένα όχημα επιθυμεί να ενημερώσει την RSU για ένα γεγονός, αν είναι στην εμβέλειά της τότε την ενημερώνει άμεσα με την παραπάνω διαδικασία. Αν δεν είναι εντός εμβέλειας, χρησιμοποιεί ένα άλλο όχημα ως σημείο πρόσβασης και το στέλνει σε αυτό, το οποίο με τη σειρά του το μεταδίδει προς την RSU. Σε περίπτωση που δεν υπάρχουν ούτε άλλα κοντινά οχήματα, τότε το όχημα περιμένει να εισέλθει στην περιοχή μίας RSU.

Η RSU, όταν λάβει πολλά μηνύματα, για να τα στείλει στο τοπικό σύννεφο έχει τη δυνατότητα να τα ομαδοποιήσει και να τα στείλει σε μεγάλα πακέτα, για την αποφυγή συμφόρησης στο δίκτυο. Το σύννεφο, από την άλλη, αν το θεωρήσει αναγκαίο θα ενημερώσει για κάποιο γεγονός τους χρήστες μέσω της RSU. Το σχήμα, επίσης, προβλέπει και για την διατήρηση της ανωνυμίας των οχημάτων με τη χρήση ψεύδο-ταυτοτήτων, που αναφέρθηκαν στα προηγούμενα σχήματα. Το σχήμα επεκτείνει την λογική του Mix-Zone για τη δημιουργία και τη διάδοση των ψευδωνύμων σε μία περιοχή, μέσω της RSU, ώστε να μην είναι εφικτό για έναν επιτιθέμενο να αντιστοιχίσει τα οχήματα με τα ψευδώνυμά τους. Τα ψευδώνυμα αλλάζουν όταν το πλήθος των οχημάτων στην περιοχή μικραίνει. Για να αποτραπεί αυτή η αντιστοίχιση, η RSU ενημερώνει κάποια οχήματα στην περιοχή, όταν η πυκνότητα έχει γίνει μικρή, να συμπεριφέρονται ως πολλά οχήματα, με σκοπό ο επιτιθέμενος να θεωρεί πως περισσότερα οχήματα λαμβάνουν μέρος στην ανταλλαγή ψευδωνύμων και να του είναι πιο δύσκολο να τα αντιστοιχίσει.

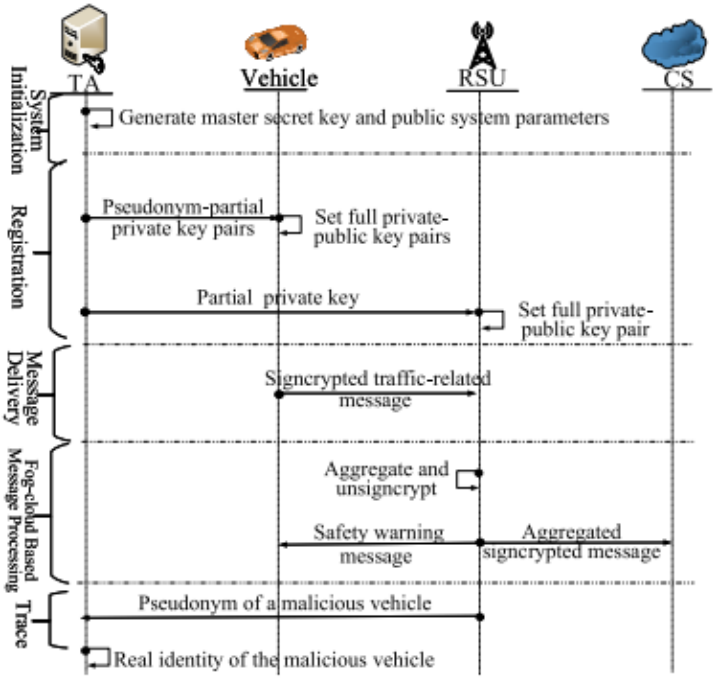


Εικόνα 12: Ανταλλαγή ψευδωνύμων σε δίκτυα οχημάτων στο σύννεφο [23]

Το παρόν σχήμα εξασφαλίζει την ασφάλεια ενάντια σε επιθέσεις τύπου Man-in-the-Middle, σε επιθέσεις επανάληψης μηνύματος και αλλοίωσης μηνύματος [18]. Επίσης, είναι αρκετά αποδοτικό μιας και τα οχήματα δεν πραγματοποιούν καμία αυθεντικοποίηση μηνυμάτων από άλλα οχήματα. Παρόλα αυτά το σχήμα δεν λαμβάνει υπόψη το μεγάλο υπολογιστικό φορτίο που μπορεί να προκύψει σε μία RSU σε περίπτωση αύξησης της πυκνότητας.

Το 2022, οι Yafang Yang κ.α. πρότειναν ένα σχήμα υπογραφής και κρυπτογράφησης μηνυμάτων σε VANETs βασισμένα στο σύννεφο, το σχήμα PPAAS [24]. Το σχήμα επιτυγχάνει ικανοποιητικό επίπεδο ασφαλείας χωρίς τη χρήση ακριβών διαδικασιών ανταλλαγής ψευδωνύμων και με δυνατότητα για επικύρωση και κρυπτογράφηση μηνυμάτων σε μεγάλα τεμάχια (batches).

Κατά τη διαδικασία εγγραφής στην Έμπιστη Αρχή, τα οχήματα ξανά λαμβάνουν τα ψευδώνυμά τους και με βάση αυτά και τα κλειδιά τους. Οι RSUs, από την άλλη, αποκτούν μόνιμα κλειδιά που δεν έχουν συγκεκριμένη χρονική ισχύ. Όταν τα οχήματα θελήσουν να στείλουν ένα μήνυμα στην RSU για κάποιο γεγονός, το μήνυμα κρυπτογραφείται και υπογράφεται ταυτόχρονα με την μέθοδο SignCrypt που παρέχει το σχήμα και στέλνεται στην RSU. Η RSU με τη χρήση του δημοσίου κλειδιού του οχήματος και του ιδιωτικού κλειδιού της αποκρυπτογραφεί και επικυρώνει το μήνυμα που έλαβε. Αν η πυκνότητα των οχημάτων στην περιοχή γίνει πολύ μεγάλη, τότε με τη χρήση ενός αναγνωριστικού ομαδοποίησης, μπορεί να αποκρυπτογραφήσει και να επικυρώσει μία ομάδα μηνυμάτων ταυτόχρονα και έτσι να μειωθεί ο συνολικός υπολογιστικός χρόνος. Ο αλγόριθμος, επίσης, προβλέπει μέθοδο για τον εντοπισμό των πραγματικών αναγνωριστικών από τα ψευδώνυμα, ώστε να αποκλείονται οι κακόβουλοι χρήστες.



Εικόνα 13: Το σχήμα PPAAS [24]

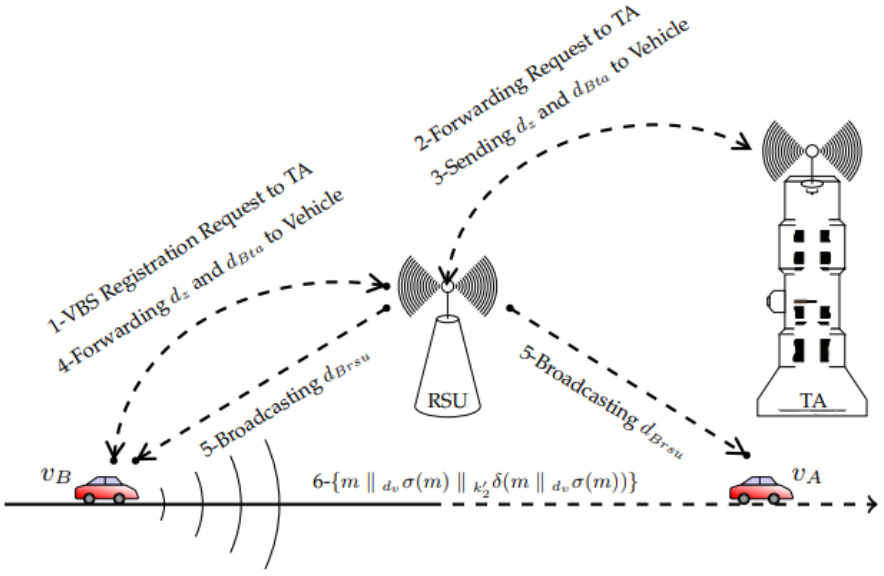
Το παραπάνω σχήμα εξασφαλίζει την ανωνυμία των οχημάτων, προσφέρει μηχανισμό εντοπισμού τυχόν κακόβουλων χρηστών, εξασφαλίζει την εμπιστευτικότητα και την αυθεντικότητα των μηνυμάτων [18]. Επίσης, βελτιώνει τον χρόνο εκτέλεσης που χρειάζεται η RSU για να επικυρώσει και να προωθήσει μηνύματα από τα οχήματα του δικτύου. Παρόλα αυτά, ο φόρτος της RSU παραμένει ακόμα μεγάλος και μπορεί να οδηγήσει σε καθυστερήσεις στο σύστημα.

Το 2021, οι Mir Ali Rezazadeh Baee κ.α. [25] πρότειναν ένα σχήμα αυθεντικοποίησης και κρυπτογράφησης μηνυμάτων σε VANETs που αξιοποιεί αρκετά χαρακτηριστικά της κρυπτογραφίας των Ελλειπτικών Καμπυλών. Το σχήμα ονομάζεται ALI: Anonymous Lightweight Inter-Vehicle Broadcast Authentication with Encryption. Στο σχήμα χρησιμοποιείται για τη διάδοση των περιστασιακών κλειδιών των οχημάτων το σχήμα ECQV που είναι βασισμένο στις Ελλειπτικές Καμπύλες. Για την ανταλλαγή συμμετρικών κλειδιών κρυπτογράφησης το σχήμα χρησιμοποιεί το ECIES, το οποίο πρακτικά χρησιμοποιεί τον μηχανισμό Diffie-Hellman για να δημιουργήσει ένα κοινό μυστικό και στη συνέχεια με τη χρήση μίας Συνάρτησης παραγωγής κλειδιών βασισμένη στον Κατακερματισμό (HKDF- Hash-based Key-Derivation Function).

Η Έμπιστη Αρχή (TA – CA) μεταβιβάζει τα κλειδιά των οχημάτων και των RSU κατά την εγγραφή τους με τη χρήση του σχήματος ECQV και τα συνδέει με ένα ψευδώνυμο – ταυτότητα που αποθηκεύει, την οποία μοιράζεται μαζί τους κρυπτογραφημένη με την τεχνική ECIES, για την γρηγορότερη επικοινωνία μιας και τα οχήματα θα χρειαστεί να στείλουν μόνο την ταυτότητα και όχι ολόκληρο το

πιστοποιητικό. Οι χρήστες λαμβάνουν τα κλειδιά τους και τα αποθηκεύουν. Τα κλειδιά έχουν ισχύ για κάποιο προκαθορισμένο χρονικό διάστημα (για παράδειγμα 6 μήνες). Όταν τα κλειδιά λήξουν, τότε οι χρήστες πρέπει να ζητήσουν από την Έμπιστη Αρχή νέα, τα οποία μεταδίδονται ξανά με τον ίδιο τρόπο.

Η επικοινωνία ενός οχήματος με την Έμπιστη Αρχή γίνεται με επώνυμο τρόπο, καθώς αποστέλλεται και η ταυτότητά του κρυπτογραφημένη. Η επικοινωνία εκτελείται μέσω της RSU. Η κρυπτογράφηση γίνεται με συμμετρικό τρόπο με την χρήση του ECIES. Για την επικοινωνία των οχημάτων με άλλα οχήματα, η RSU είναι υπεύθυνη να μεταδίδει ανά κάποιο χρονικό διάστημα ένα συμμετρικό κλειδί, που θα χρησιμοποιηθεί για την κρυπτογράφηση, σε όλα τα οχήματα της περιοχής. Για να είναι ασφαλής η μετάδοση, η παραγωγή του συμμετρικού κλειδιού γίνεται μόνο με τη χρήση ενός δημοσίου κλειδιού από την Έμπιστη Αρχή. Επομένως, το σχήμα απαιτεί την μία «ημερήσια χειραψία» των οχημάτων με την Έμπιστη Αρχή, ώστε να λαμβάνουν το δημόσιο κλειδί ανά 24 ώρες. Κάθε περιοχή, που μπορεί να συμπεριλαμβάνει πολλές RSU, λαμβάνει ένα δημόσιο κλειδί από την Έμπιστη Αρχή ανά 24 ώρες. Επιπλέον, τα μηνύματα που στέλνονται πρέπει να περιέχουν την ταυτότητα του οχήματος κρυπτογραφημένη με ECIES με το δημόσιο κλειδί της Έμπιστης Αρχής, καθώς σε περίπτωση διαφωνίας, ένα όχημα στέλνει το εν λόγω μήνυμα στην Έμπιστη Αρχή κρυπτογραφημένο και η Έμπιστη Αρχή μονάχα μπορεί να εντοπίσει την πραγματική ταυτότητα του οχήματος που έστελνε μηνύματα με πιθανόν κακόβουλη πρόθεση.



Εικόνα 14: V2V επικοινωνία στο σχήμα ALI [25]

Το παραπάνω σχήμα εισάγει για πρώτη φορά τη χρήση του σχήματος ECQV για τη μετάδοση των κλειδιών σε VANETs. Επίσης, αξιοποιεί πλήρως τις κρυπτογραφικές τεχνικές των Ελλειπτικών Καμπυλών πράγμα που δημιουργεί μικρότερα κλειδιά και επομένως μηνύματα μικρότερου μεγέθους στο δίκτυο. Το σχήμα, επιπλέον, προβλέπει για την αυθεντικοποίηση, την ιδιωτικότητα ταυτότητας, την Ανάκληση

των πιστοποιητικών των κακόβουλων χρηστών, την αδυναμία αποκήρυξης κάποιας πράξης και την ασφαλή ανανέωση των κλειδιών [18]. Παρόλα αυτά, η επικύρωση των μηνυμάτων γίνεται από τα ίδια τα οχήματα και όχι από την RSU, πράγμα που μπορεί να δημιουργήσει καθυστερήσεις, αφού τα οχήματα έχουν μικρή υπολογιστική δύναμη.

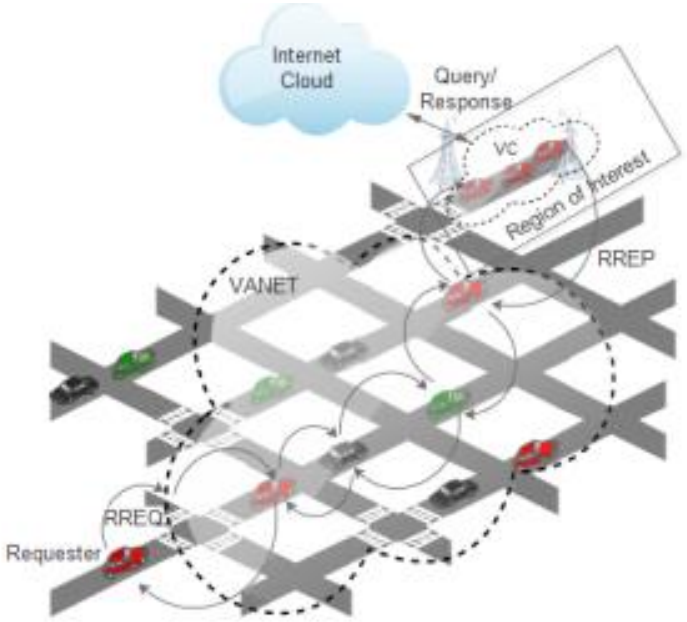
3.1.3 Σχήματα κατανομής του φόρτου στο δίκτυο

Παραπάνω, αναλύθηκαν διάφορα προτεινόμενα σχήματα αυθεντικοποίησης και ασφαλής διάδοσης μηνυμάτων σε VANETs, με σκοπό την διασφάλιση της ασφαλούς επικοινωνίας. Μερικά από αυτά λαμβάνουν υπόψη τους πως η υπολογιστική ισχύ των οχημάτων είναι περιορισμένη και πως το εύρος ζώνης στα VANETs είναι περιορισμένο. Πιο σύγχρονα σχήματα έχουν δημιουργηθεί με σκοπό να βελτιωθούν οι επιδόσεις σε χρόνο υπολογισμού και εύρους ζώνης, στην καλύτερη διαχείριση των πόρων, χωρίς να μειώνεται η ασφάλεια τους. Για να το πετύχουν αυτό, αξιοποιούν τεχνικές που αφορούν τα Δίκτυα που δεν βασίζονται στην υποδομή (δηλαδή στις RSU), αλλά επικοινωνούν απευθείας με το σύννεφο ή με την Έμπιστη Αρχή. Διάφορα σχήματα βασισμένα στην ομαδοποίηση των οχημάτων με βάση την γεωγραφική τους θέση έχουν αναπτυχθεί στη σύγχρονη βιβλιογραφία. Κάποια από αυτά αντιμετωπίζουν και ζητήματα ασφαλείας.

Για την καλύτερη κατανόηση των τεχνικών ομαδοποίησης, γίνεται αναφορά στο σχήμα των Moumena Chaqfeh κ.α. [26] που προτάθηκε το 2016 με σκοπό την ομαδοποίηση των οχημάτων σε VANETs βασισμένα στο σύννεφο (cloud-based). Το συγκεκριμένο μοντέλο δεν αντιμετωπίζει το ζήτημα της ασφάλειας, ωστόσο η αναφορά του αποσκοπεί στην καλύτερη κατανόηση των επόμενων σχημάτων ασφαλείας που αναφέρονται στο παρόν κεφάλαιο. Σε αυτό το σχήμα, τα οχήματα που βρίσκονται σε μία περιοχή δημιουργούν ένα σύννεφο και συνεργάζονται, ώστε να απαντήσουν σε αιτήματα που καταφθάνουν στην περιοχή από άλλα οχήματα. Η μετάδοση των αιτημάτων γίνεται με τη χρήση αλγόριθμου δρομολόγησης (για παράδειγμα AODV). Όταν το αίτημα καταφθάσει στην περιοχή ενδιαφέροντος, τότε τα οχήματα συνεργάζονται και απαντούν στο αίτημα. Για να συντονιστεί ο κατανεμημένος υπολογισμός απαιτείται να υπάρξει μία περισυλλογή όλων των αποτελεσμάτων και να γίνει αποστολή της απάντησης. Αυτή τη διαδικασία την αναλαμβάνει ένα από τα οχήματα της περιοχής που εκλέγεται από την RSU ως μεσίτης ή broker. Αυτός αναλαμβάνει να περισυλλέξει τα αποτελέσματα και να απαντήσει στο αίτημα, ή αν είναι αναγκαία η πιο περίπλοκη επεξεργασία τους να τα αποστείλει στο σύννεφο. Από εκεί και πέρα, ο μεσίτης της περιοχής από την οποία προήλθε το αίτημα, αναλαμβάνει να συλλέξει την απάντηση του αιτήματος και να την μεταβιβάσει στο όχημα της περιοχής που το εκτέλεσε.

Η παρούσα υλοποίηση καταφέρνει να ομαδοποιήσει τα οχήματα σε γεωγραφικές περιοχές με αποτελεσματικότητα και να θεσπίσει έναν μηχανισμό

μεταβίβασης αιτημάτων, μέσω ενός κεντρικού μεσίτη σε κάθε περιοχή, ο οποίος εκλέγεται από την RSU.

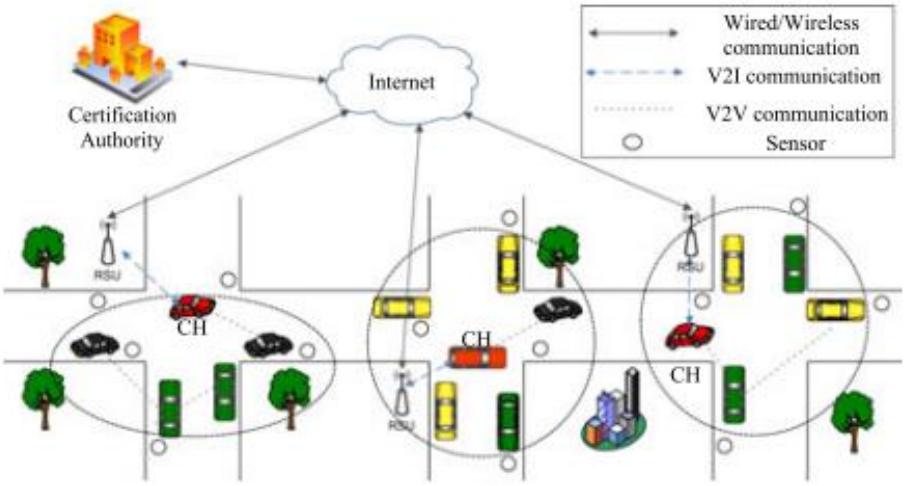


Εικόνα 15: Γεωγραφική ομαδοποίηση οχημάτων σε Vehicular Cloud δίκτυα [26]

Με παρόμοιο τρόπο, το 2018 οι Amit Dua κ.α. [27] προσπάθησαν να ομαδοποιήσουν τη διαδικασία αυθεντικοποίησης των οχημάτων με τη χρήση των Κεφαλών Ομάδας (Cluser Heads). Παρόμοια με πριν, τα οχήματα δημιουργούν ομάδες ανάλογα με διάφορα κριτήρια, όπως είναι η γεωγραφική θέση, η ταχύτητα και η υπολογιστική ισχύ τους. Ένα από τα οχήματα που βρίσκεται στην ομάδα εκλέγεται ως Cluster Head, ή εν συντομία CH. Ο CH αυθεντικοποιείται από την Έμπιστη Αρχή και λαμβάνει μία ομαδική ταυτότητα (group ID – GID) που θα χρησιμοποιηθεί αργότερα από τα οχήματα της ομάδας του για αυθεντικοποίηση και ανταλλαγή κλειδιών.

Για να γίνει η αυθεντικοποίηση οποιουδήποτε CH, η Έμπιστη Αρχή εγγράφει τα οχήματα στη βάση της πριν την κυκλοφορία τους και τα εφοδιάζει με κάποιες παραμέτρους, όπως είναι δημόσια κλειδιά και υπογραφές. Τα δημόσια κλειδιά και οι υπογραφές είναι βασισμένες στις Ελλειπτικές Καμπύλες και η αυθεντικοποίηση επιτυγχάνεται με την τεχνική ElGamal. Αφού αυθεντικοποιηθεί ο CH, αναλαμβάνει να αυθεντικοποιήσει τα υπόλοιπα οχήματα της περιοχής που διαχειρίζεται, με βάση το GID που το παρείχε η Έμπιστη Αρχή. Τα οχήματα, στη συνέχεια, πραγματοποιούν μία ανταλλαγή ενός συμμετρικού κλειδιού συνεδρίας (Session Key), που θα χρησιμοποιήσουν για την περαιτέρω επικοινωνία μεταξύ τους.

Το σχήμα αποδεικνύεται πως παρέχει εμπιστευτικότητα με τη χρήση της κρυπτογραφίας Ελλειπτικών Καμπυλών, εξασφαλίζει την αυθεντικοποίηση και είναι ανεκτικό ενάντια σε επιθέσεις Man-in-the-Middle, ενώ εξασφαλίζει και την ιδιωτικότητα ταυτότητας [18]. Επιπλέον, μειώνει τους χρόνους υπολογισμού και βελτιώνει το μέγεθος και τον αριθμό των απαιτούμενων μηνυμάτων.



Εικόνα 16: Αυθεντικοποίηση βασισμένη στους CH [27]

Το 2022, οι Hassan Mistareehi και D. Manivannan [28], οι οποίοι πρότειναν και το σχήμα που αναφέρθηκε προηγουμένως για την ασφαλή διάδοση μηνυμάτων σε δίκτυα οχημάτων στο σύννεφο (vehicular cloud), έρχονται να συνδυάσουν το παραπάνω σκεπτικό της ομαδοποιημένης αυθεντικοποίησης με τα σχήματα ασφαλούς αυθεντικοποίησης και διάδοσης μηνυμάτων, με σκοπό να διαμοιράσουν το υπολογιστικό φορτίο των RSU στο δίκτυο και παράλληλα να περιορίσουν τους υπολογισμούς στα οχήματα.

Στο συγκεκριμένο σχήμα, η RSU είναι υπεύθυνη για να επικυρώνει την αυθεντικότητα και την ακεραιότητα των μηνυμάτων πριν τα διανείμει σε άλλες RSU ή πριν ενημερώσει τα οχήματα της περιοχής για ένα γεγονός. Επιπρόσθετα, η RSU έχει τη δυνατότητα να επιλέξει ένα όχημα της περιοχής για να εκτελέσει τον ρόλο του Αρχηγού Ομάδας, ένας ρόλος αντίστοιχος του CH του προαναφερθέντος σχήματος. Η RSU γνωρίζει για την πυκνότητα των οχημάτων στην περιοχή και που σε ποια σημεία είναι μεγάλη, εφόσον γνωρίζει την γεωγραφική θέση κάθε οχήματος. Έτσι, με έναν απλό αλγόριθμο μπορεί να επιλέξει πότε η πυκνότητας μίας υπό-περιοχής έχει γίνει μεγάλη και έτσι να επιλέξει έναν GL που θα διαχειρίζεται την αυθεντικοποίηση και την ομαδοποίηση των μηνυμάτων των οχημάτων σε μεγάλες παρτίδες (batches). Έτσι, η RSU θα λαμβάνει τα ομαδοποιημένα μηνύματα και θα χρειάζεται να αυθεντικοποιήσει μονάχα τον GL για να επιβεβαιώσει την πληροφορία, πράγμα που μειώνει σημαντικά το υπολογιστικό φορτίο της. Ο GL λαμβάνει τον ρόλο της RSU στην υπό-περιοχή του και αυθεντικοποιεί τα οχήματα, τα οποία προσπαθούν να μεταδώσουν κάποια πληροφορία. Σε μία περιοχή εμβέλειας μιας RSU μπορεί να έχουν εκλεχθεί περισσότεροι από έναν GL και έτσι ο φόρτος από την RSU θα μοιραστεί στις πολύ πυκνές περιοχές. Από την άλλη, οι GL

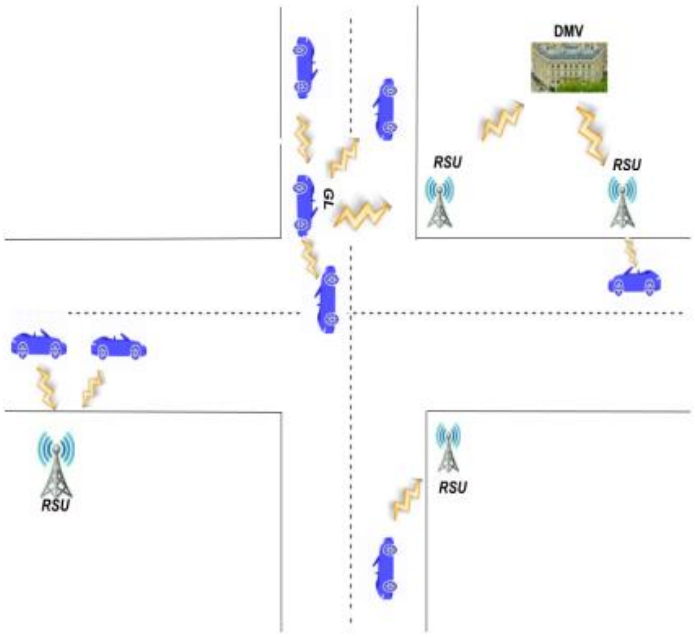
εκλέγονται μόνο όταν η πυκνότητα έχει γίνει μεγάλη και επομένως δεν επιβαρύνονται σημαντικά τα οχήματα σε γενικές γραμμές.

Για αρχή, όταν ένα όχημα εισέρχεται στην περιοχή μίας RSU λαμβάνει το πιστοποιητικό της, το οποίο αναμεταδίδει η RSU στην περιοχή περιοδικά. Το όχημα, εφοδιασμένο με το δημόσιο κλειδί της Έμπιστης Αρχής (εδώ αναφέρεται ως DMV, για λόγους ομοιομορφίας θα αναφέρεται ως Trusted ή Certificate Authority), καταφέρνει να ανακτήσει από το πιστοποιητικό το δημόσιο κλειδί της RSU και να σιγουρευτεί πως είναι έγκυρο. Με χρήση ασύμμετρης κρυπτογράφησης, το όχημα στέλνει ένα μήνυμα "Join", μαζί με το πιστοποιητικό του, στην RSU κρυπτογραφημένο με το δημόσιο κλειδί της και υπογεγραμμένο με το ιδιωτικό κλειδί του οχήματος. Το ζεύγος κλειδιών του οχήματος το παρέχει η Έμπιστη Αρχή – CA κατά την εγγραφή των οχημάτων. Η RSU ανακτά το δημόσιο κλειδί του οχήματος, επικυρώνει την υπογραφή με αυτό και ελέγχει την χρονική στάμπα. Αν αυτές οι ενέργειες ολοκληρωθούν με επιτυχία, η RSU αποδέχεται το όχημα στην περιοχή, κρυπτογραφεί ένα συμμετρικό κλειδί και το στέλνει πίσω στο όχημα κρυπτογραφημένο με το δημόσιο κλειδί του, μαζί με ένα μήνυμα "Accept". Το όχημα αποθηκεύει το συμμετρικό κλειδί για μελλοντική επικοινωνία.

Σε περίπτωση που η RSU λαμβάνει πολλά μηνύματα "Join" από κάποια περιοχή, εκλέγει σε εκείνη έναν GL και τον εφοδιάζει με ένα «αποδεικτικό αρχηγίας» (Proof of Leadership). Αυτό το αποδεικτικό χρησιμοποιείται από τον GL, για να αποδείξει ότι είναι ο εκλεγμένος αρχηγός από την RSU στα άλλα οχήματα. Ο GL, στη συνέχεια, αναμεταδίδει ανά κάποιο χρονικό διάστημα το αποδεικτικό αρχηγίας και το δημόσιο κλειδί του. Τα οχήματα που εισέρχονται στην περιοχή του, εκτελούν την διαδικασία "Join" ξανά, αλλά μαζί του, αφού επιβεβαιώσουν την γνησιότητά του και έτσι ανταλλάσσουν ένα συμμετρικό κλειδί. Το συμμετρικό κλειδί χρησιμοποιείται για την κρυπτογράφηση των μηνυμάτων που στέλνουν τα οχήματα στον GL ή στην RSU. Ο GL έχει την υποχρέωση να επικυρώνει τα μηνύματα και να τα ομαδοποιεί και ανά περιόδους στέλνει τα ομαδοποιημένα μηνύματα στην RSU σε batches. Η RSU επικυρώνει το λαμβανόμενο μήνυμα από τον GL και ανακτά την πληροφορία αποκρυπτογραφώντας το με το συμμετρικό κλειδί. Στη συνέχεια, εκτελεί τις απαιτούμενες ενέργειες (διάδοση σε άλλες RSU, ενημέρωση της περιοχής για κάποιο γεγονός).

Η RSU πρέπει, επιπλέον, να μεταδίδει την Λίστα Ανάκλησης στα οχήματα, την οποία λαμβάνει από την Έμπιστη Αρχή. Η Έμπιστη Αρχή αναλαμβάνει να ανανεώνει την Λίστα και να αποκλείει τους κακόβουλους χρήστες. Αν έχει εκλεγεί κάποιος GL, τότε η Λίστα Ανάκλησης μεταδίδεται στην περιοχή ευθύνης του GL μέσω αυτού. Επιπλέον, οι GLs και η RSU πρέπει σε περίπτωση εντοπισμού ενός κακόβουλου χρήστη να ενημερώσουν την Έμπιστη Αρχή, ωστόσο η συγκεκριμένη διαδικασία δεν αφορά το συγκεκριμένο σχήμα. Επιπλέον, η ανωνυμία των οχημάτων εξασφαλίζεται ξανά με ψευδώνυμα. Κάθε μήνυμα που μεταδίδεται φέρει και το ψευδώνυμο του οχήματος ή της RSU που το στέλνει. Η ανανέωση και μετάδοση των ψευδωνύμων στην περιοχή μπορεί να γίνει με οποιονδήποτε αλγόριθμο προαναφέρθηκε στο

κεφάλαιο, ωστόσο, οι συγγραφείς προτείνουν το σχήμα που είχαν υλοποίηση στην δημοσίευση τους για την ασφαλή διάδοση μηνυμάτων σε δίκτυα οχημάτων στο σύννεφο, η οποία επεκτείνει την λειτουργία Mix-Group. Η ανανέωση των ζευγών κλειδιών, επίσης, γίνεται ανά κάποιο χρονικό διάστημα από την Έμπιστη Αρχή, παρόμοια με το σχήμα ALI.



Εικόνα 17: Το σχήμα ασφαλείας με τη χρήση Group Leader [28]

Το σχήμα αξιοποιεί αρκετές από τις τεχνικές που προαναφέρθηκαν για την εξασφάλιση και βελτίωση της εμπιστευτικότητας, ακεραιότητας και της αυθεντικότητας των δεδομένων. Το σχήμα είναι ασφαλές από επιθέσεις Impersonation, Man-in-the-Middle, Replay με τη χρήση χρονικής στάμπας, διασφαλίζει την ανωνυμία με τη χρήση των ψευδωνύμων [18], αλλά και μειώνει αποτελεσματικά το γενικό κόστος υπολογισμών και επικοινωνίας κατανέμοντας τον φόρτο σε πολλά μέρη του δικτύου, όταν αυτό γίνει αναγκαίο. Αξιοποιεί τεχνικές ασύμμετρης και συμμετρικής κρυπτογραφίας και αφήνει ανοιχτή την επιλογή των αλγόριθμων κρυπτογράφησης που θα χρησιμοποιηθούν. Το συγκεκριμένο σχήμα επιλέχθηκε για την διεκπεραίωση της προσομοίωσης με σκοπό την λύση του προβλήματος που αναφέρεται στο επόμενο κεφάλαιο.

3.1.4 Πίνακας σύγκρισης

Πίνακας 2: Πίνακας Σύγκρισης σχημάτων ασφαλείας σε VANETs

Σχήμα	Πρόβλημα	Κρυπτογραφικές τεχνικές	Ασφάλεια	Επίδοση
[20]	Αυθεντικοποίηση σε VANETs	Bilinear Pairing στο Z^*_q	Εξασφάλιση αυθεντικότητας – ασφαλές ενάντια σε Impersonation – Message Tampering - Ιδιωτικότητα ταυτότητας	Αποδοτικοί υπολογισμοί, όμως η αυθεντικοποίηση γίνεται στα οχήματα
[21]	Αυθεντικοποίηση δύο σταδίων σε VANETs	Bilinear Pairing στο Z^*_q, MAC, timestamps	Εξασφάλιση αυθεντικότητας – ασφαλές ενάντια σε Impersonation – Message Tampering – Replay Attack - Ιδιωτικότητα ταυτότητας	Χρονοβόροι υπολογισμοί, μεγάλο κόστος επικοινωνίας, αυθεντικοποίηση πάντα μέσω TA
[22]	Συνεργατική αυθεντικοποίηση σε VANETs	Bilinear Pairing και Point Multiplication σε ECC, Hashing, ECDSA, AES	Εξασφάλιση αυθεντικότητας – ασφαλές ενάντια σε Impersonation – Message Tampering – Replay Attack - Ιδιωτικότητα ταυτότητας – Διαμοιρασμός RL	Σχετικά αποδοτικοί υπολογισμοί, μεγάλα μηνύματα, αυθεντικοποίηση στα οχήματα και στην RSU
[23]	Αυθεντικοποίηση + Ασφαλής διάδοση μηνυμάτων σε VANETs	Συμμετρική και Ασύμμετρη Κρυπτογράφηση, Ψηφιακές υπογραφές, Ψευδώνυμα Mix-Zone, timestamps	Εξασφάλιση αυθεντικότητας και εμπιστευτικότητας – ασφαλές ενάντια σε Man-in-the-Middle – Replay – Message modification – Ιδιωτικότητα ταυτότητας/θέσης	Αποδοτικοί υπολογισμοί, λίγα μηνύματα, αυθεντικοποίηση και επεξεργασία στην RSU, υπερφόρτωση RSU σε μεγάλες πυκνότητες
[24]	Αυθεντικοποίηση + ασφαλής διάδοση μηνυμάτων σε VANETs	Signcryption με Bilinear Maps στο Z^*_q, Ψευδώνυμα, Hashing	Εξασφάλιση αυθεντικότητας και εμπιστευτικότητας – Ιδιωτικότητα ταυτότητας	Αποδοτικοί υπολογισμοί, αυθεντικοποίηση και επεξεργασία στην RSU σε batches

[25]	Αυθεντικοποίηση + ασφαλής διάδοση μηνυμάτων σε VANETs	ECIES, Diffie-Hellman, ECQV, ECDSA, HKDF	Εξασφάλιση αυθεντικότητας και εμπιστευτικότητας – ασφαλές ενάντια σε Man-in-the-Middle – Replay – Message modification – Ιδιωτικότητα ταυτότητας/θέσης - Revocation	Αποδοτικοί υπολογισμοί, μικρά μηνύματα, αυθεντικοποίηση και επεξεργασία στα οχήματα και στην RSU
[26]	Ομαδοποίηση και συνεργασία σε cloud-based VANETs	-	-	Συνεργασία οχημάτων, μεγαλύτερος φόρτος στον broker
[27]	Αυθεντικοποίηση + ασφαλής διάδοση μηνυμάτων σε VANETs με κατανεμημένο τρόπο	ECC ElGamal signatures, ασύμμετρη κρυπτογράφηση ECC, συμμετρική κρυπτογράφηση AES	Εξασφάλιση αυθεντικότητας και εμπιστευτικότητας – ασφαλές ενάντια σε Man-in-the-Middle – Ιδιωτικότητα ταυτότητας/θέσης	Αποδοτικοί υπολογισμοί, μικρά μηνύματα, αυθεντικοποίηση και επεξεργασία στα οχήματα και στον CH
[28]	Αυθεντικοποίηση + ασφαλής διάδοση μηνυμάτων σε VANETs με κατανεμημένο τρόπο	Ασύμμετρη κρυπτογράφηση RSA, Συμμετρική κρυπτογράφηση AES, ψηφιακές υπογραφές και πιστοποιητικά δημοσίου κλειδιού, Ψευδώνυμα, Hashing	Εξασφάλιση αυθεντικότητας και εμπιστευτικότητας – ασφαλές ενάντια σε Man-in-the-Middle – Replay – Message modification – Ιδιωτικότητα ταυτότητας/θέσης - Revocation	Αποδοτικοί υπολογισμοί, μικρά μηνύματα, αυθεντικοποίηση και επεξεργασία στην RSU και στους GL όταν υπάρχει μεγάλη πυκνότητα

Κεφάλαιο 4: Περιγραφή του Προβλήματος

Στο παρόν κεφάλαιο παρουσιάζεται το πρόβλημα το οποίο πραγματεύεται η διπλωματική εργασία, η βασική ιδέα για την αντιμετώπισή του και ο γενικότερος σκοπός της.

4.1 Παρουσίαση του προβλήματος

Τα Αυτό-Οργανούμενα Δίκτυα Οχημάτων (VANETs) έχουν λάβει σημαντική αναγνώριση τα τελευταία χρόνια, καθώς μέσω αυτών μπορεί να έρθει η επανάσταση στα έξυπνα συστήματα μεταφοράς (ITS). Παρόλα αυτά, η ιδιαίτερη φύση τους δημιουργεί νέες προκλήσεις στον τομέα της ασφάλειας και της ιδιωτικότητας. Τα ζητήματα αυτά είναι ύψιστης σημασίας στα VANETs, αφού η διαρροή των δεδομένων ή της θέσης ενός οχήματος και οι κακόβουλοι χρήστες που υποδύονται άλλα οχήματα, μπορούν να θέσουν τους οδηγούς σε κίνδυνο. Εφόσον τα VANETs, από τη φύση τους, χαρακτηρίζονται ως δυναμικά και περιορισμένων πόρων είναι απαραίτητα τα σχήματα ασφάλειας και ιδιωτικότητας να είναι όσο το δυνατόν πιο αποδοτικά.

Ιδιαίτερα τα τελευταία χρόνια, μεγάλο μέρος της έρευνας των VANETs αφοσιώνεται στη δημιουργία ενός πιο ασφαλέστερου οδικού δικτύου. Πολλοί αλγόριθμοι έχουν προταθεί για την αποφυγή ατυχημάτων, τη διαχείριση της κίνησης και για υπηρεσίες έκτακτης ανάγκης. Ωστόσο, εφόσον τα VANETs είναι βασισμένη στην ασύρματη και ανοιχτή επικοινωνία, είναι αρκετά ευάλωτα σε κινδύνους ασφάλειας. Οι κακόβουλοι χρήστες μπορούν να αξιοποιήσουν διάφορες τεχνικές για να δημιουργήσουν σύγχυση στο δίκτυο, μέχρι και τροχαία ατυχήματα. Κάποιες από αυτές είναι οι επιθέσεις Impersonation, Message Tampering, Bogus Message [18] και να εντοπίσουν την τοποθεσία των οχημάτων δημιουργώντας κενά ασφαλείας.

Η ιδιωτικότητα των δεδομένων είναι επίσης ένα πολύ σημαντικό κομμάτι που η επιστημονική κοινότητα καλείται να αντιμετωπίσει. Οι εφαρμογές που αποσκοπούν στην ασφάλεια των οδηγών, απαιτούν από τα οχήματα να μοιράζονται σε συχνή βάση δεδομένα για την ταχύτητα, τη θέση και την οδηγική συμπεριφορά. Αν αυτά τα δεδομένα δεν είναι επαρκώς ασφαλισμένα, τότε κακόβουλοι χρήστες μπορούν να τα εκμεταλλευτούν για να εντοπίζουν οχήματα ή να παραβιάσουν περαιτέρω προσωπικά στοιχεία [18].

Η αποδοτικότητα των σχημάτων ασφάλειας και προστασίας της ιδιωτικότητας είναι και αυτό ένα μείζον ζήτημα, αφού πολλά από τα ήδη υπάρχοντα σχήματα ασφαλείας στο διαδίκτυο δεν είναι κατάλληλα να εφαρμοστούν στα VANETs, τα οποία διαθέτουν περιορισμένους πόρους. Το εύρος ζώνης, η κατανάλωση ενέργειας, ο χρόνος υπολογισμού, η δυναμική φύση των VANETs, όπου

πολλοί νέοι χρήστες εισέρχονται και εξέρχονται σε μικρό χρονικό διάστημα στο δίκτυο είναι από τα πιο σημαντικά ζητήματα.

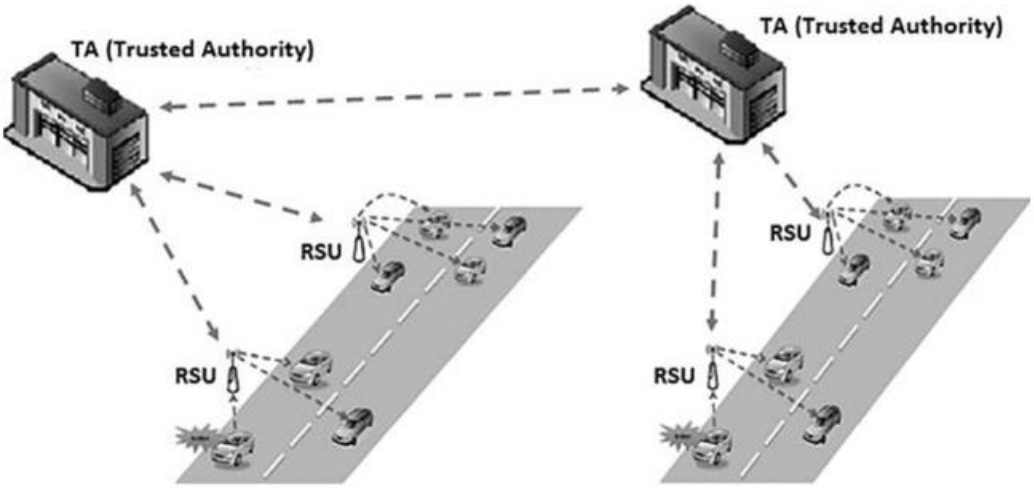
Μερικά από αυτά τα σχήματα αναλύθηκαν στο προηγούμενο κεφάλαιο και κάποια από αυτά λαμβάνουν υπόψη τους την απόδοση. Παρατηρείται συνεχής βελτίωση στον τομέα της επίδοσης των σχημάτων ασφαλείας, όμως ακόμα χρήζουν μεγαλύτερης επίδοσης. Γι' αυτό η συγκεκριμένη διπλωματική εργασία αποσκοπεί στη χρήση εναλλακτικών τεχνικών για την βελτίωση της επίδοσης στα VANETs. Μία από τις υποσχόμενες τεχνικές που επανήλθε στην επιφάνεια με την εξέλιξη του IoT είναι η κρυπτογραφία Υπερελλειπτικών Καμπυλών, η οποία έχει ιδιότητες που ταιριάζουν στο περιβάλλον των VANETs.

4.2 Μοντέλο δικτύου

Για την καλύτερη κατανόηση του προβλήματος και για την καλύτερη οργάνωση της υλοποίησης της λύσης του τα VANETs και οι χρήστες τους θεωρούνται πως έχουν την εξής μορφή:

- Έμπιστη Αρχή (CA – TA): Η Έμπιστη Αρχή είναι ένας επίσημος οργανισμός, ο οποίος είναι υπεύθυνος να εγγράφει τα οχήματα και να τα εφοδιάζει με τα κατάλληλα εργαλεία, ώστε να είναι επικυρωμένα ως γνήσια/μη κακόβουλα όταν εισέρχονται σε ένα δίκτυο. Διαχειρίζεται τα πιστοποιητικά, τα κλειδιά και τις Λίστες Ανάκλησης και είναι υπεύθυνη να τα ανανεώνει όταν αυτό είναι αναγκαίο. Έχει τη δυνατότητα να αποκλείει κακόβουλους χρήστες από τα δίκτυα, ενώ μπορεί να του ζητηθεί να αυθεντικοποιήσει οχήματα ή RSU. Η Έμπιστη Αρχή μπορεί να είναι κάποιο υπουργείο μεταφορών ή κάποια άλλη επίσημη αρχή και θεωρείται έμπιστη από όλους τους χρήστες των δικτύων. Επικοινωνεί με τις RSU μέσω ασφαλών καναλιών επικοινωνίας.
- Μονάδα Υποδομής (RSU): Είναι μία στατική μονάδα επικοινωνίας και τοποθετείται σε ένα σταθερό σημείο στο δίκτυο. Αναλαμβάνει να καλύψει μία περιοχή και έχει ένα σύνολο από αρμοδιότητες, όπως να συλλέγει πληροφορίες, να ενημερώνει τα οχήματα για την κατάσταση της περιοχής και σε περιπτώσεις έκτακτης ανάγκης, να μεταφέρει μηνύματα στην Έμπιστη Αρχή ή σε άλλες RSU, ενώ σε πολλές περιπτώσεις αναλαμβάνει να αυθεντικοποιεί τους χρήστες και να εξασφαλίζει την ασφαλή επικοινωνία και την ιδιωτικότητα. Οι RSU είναι πιο δαπανηρές συσκευές και έτσι τοποθετούνται αραιά στο τοπικό δίκτυο, ενώ έχουν μεγαλύτερη υπολογιστική ισχύ και, εφόσον είναι μέρος της υποδομής, μεγαλύτερες ενεργειακές δυνατότητες. Οι RSU επικοινωνούν συνήθως με το πρωτόκολλο WAVE που είναι βασισμένο στο WiFi (αναλύθηκε στο 2^{ο} κεφάλαιο).
- Όχημα: Είναι ένα όχημα εφοδιασμένο με ειδικό εξοπλισμό, ώστε να έχει τη δυνατότητα να επικοινωνήσει και να επεξεργαστεί δεδομένα με άλλα οχήματα (V2V) ή με την υποδομή (V2I). Τα οχήματα εφοδιάζονται με On-

Board Units (OBU) που προσφέρει τη δυνατότητα επικοινωνίας με το πρωτόκολλο WAVE. Η OBU στις περισσότερες περιπτώσεις αναμεταδίδει ανά κάποιο χρονικό διάστημα πληροφορία σχετικά με τη θέση, την ταχύτητα, την επιτάχυνση κ.α. στο δίκτυο. Για την ασφαλή επικοινωνία, η OBU από μόνη της διαθέτει έναν καταγραφέα δεδομένων και μία αμετάβλητη συσκευή (Tamper-Proof Device – TPD). Το TPD αναλαμβάνει να αποθηκεύει με ασφάλεια και τα κρυπτογραφικά διαπιστευτήρια του οχήματος (κλειδιά, πιστοποιητικά, υπογραφές) και την εκτέλεση των κρυπτογραφικών υπολογισμών, όπως κρυπτογράφηση, υπογραφή, επικύρωση κ.α. Το TPD απαιτεί δική του μπαταρία και ρολόι για την ομαλή λειτουργία του.



Εικόνα 18: Μοντέλο Δικτύου (Από: https://ebrary.net/183106/computer_science/communication_vanet)

4.3 Μοντέλο επιτιθέμενων

Η παρούσα εργασία λαμβάνει υπόψη τους παρακάτω τύπους επιτιθέμενων:

- **Καθολικός παθητικός**: Είναι εξωτερικός κακόβουλος χρήστης, ο οποίος κρυφακούει την επικοινωνία που λαμβάνει χώρα σε όλο το δίκτυο και υποκλέπτει μηνύματα και παρακολουθεί όλα τα οχήματα.
- **Τοπικός παθητικός**: Είναι εξωτερικός κακόβουλος χρήστης, ο οποίος κρυφακούει και υποκλέπτει μηνύματα στην περιοχή μόνο μίας RSU. Επομένως, αποτελεί κίνδυνο για την περιοχή εμβέλειας μίας RSU.
- **Εσωτερικός κατάσκοπος**: Είναι ένας εσωτερικός κακόβουλος χρήστης, ο οποίος μεταμφιέζεται σε κάποιο έγκυρο μέλος του δικτύου και συμμετέχει κανονικά στην επικοινωνία. Ωστόσο, αποσκοπεί στην κλοπή χρήσιμων πληροφοριών τις οποίες μοιράζεται εκτός του δικτύου.
- **Εσωτερικός ταραχοποιός**: Είναι εσωτερικός επιτιθέμενος και αποσκοπεί στη δημιουργία χάους στο δίκτυο.

Επιθέσεις στο ίδιο το υλικό και το λογισμικό των οχημάτων ή της υποδομής του δικτύου δεν λαμβάνονται υπόψη. Η εργασία και τα σχήματα ασφαλείας περιορίζονται σε χρήστες που έχουν πρόσβαση στο δίκτυο. Ωστόσο, ένα επιτιθέμενος που θα είχε τη δυνατότητα να παρέμβει στο λογισμικό ή υλικό ενός χρήστη θα ήταν πολύ αποτελεσματικός.

Οι πιο σημαντικές επιθέσεις που πρέπει να αποφευχθούν και λαμβάνει υπόψη η εργασία είναι:

- **Man-in-the-Middle Attack**: Ένας κακόβουλος χρήστης παρεμβάλλεται στην επικοινωνία μεταξύ δύο χρηστών του δικτύου (για παράδειγμα μεταξύ του οχήματος και της RSU ή της Έμπιστης Αρχής). Εγκαθιδρύει συνόδους επικοινωνίας με τα 2 μέρη, τα οποία νομίζουν πως επικοινωνούν μεταξύ τους, ωστόσο στην πραγματικότητα επικοινωνούν με τον επιτιθέμενο. Ο επιτιθέμενος υποχρεώνει τα 2 μέρη να χρησιμοποιήσουν τα δικά του κλειδιά για την επικοινωνία, με αποτέλεσμα να υποκλέπτει όλη την πληροφορία [18].
- **Impersonation Attack**: Ο επιτιθέμενος υποδύεται ένα έγκυρο όχημα ή RSU, για να δημιουργήσει σύγχυση με σκοπό να κερδίσει πρόσβαση σε ευαίσθητα δεδομένα [18].
- **Replay Attack**: Ο επιτιθέμενος αποθηκεύει έγκυρα μηνύματα και τα αναμεταδίδει μετά από κάποιο χρονικό διάστημα με σκοπό να κερδίσει πρόσβαση σε κλειδιά ή μηνύματα [18].
- **Cryptanalysis Attack**: Ο επιτιθέμενος χρησιμοποιεί αδυναμίες των κρυπτογραφικών μεθόδων για να ανακτήσει τα κρυπτογραφικά κλειδιά και να αποκτήσει πρόσβαση σε ευαίσθητες πληροφορίες [18].
- **Message Tampering**: Ο επιτιθέμενος κρυφακούει την επικοινωνία και αποθηκεύει τα μηνύματα. Στη συνέχεια μεταβάλλει το περιεχόμενό τους με σκοπό να προκαλέσει σύγχυση και να υποκλέψει ευαίσθητες πληροφορίες. Συνδυάζεται με τις επιθέσεις Man-in-the-Middle και Impersonation [18].
- **Bogus Message Attack**: Είναι μία επίθεση ιδιαίτερα επικίνδυνη στα VANETs. Ο επιτιθέμενος επικοινωνεί στο δίκτυο (οχήματα και RSU) ψευδείς πληροφορίες, όπως για τροχαία ατυχήματα ή για την τοποθεσία και την ταχύτητά του ή κάποιου άλλου οχήματος, με σκοπό να προκαλέσει χάος [18].

4.4 Μετρικές αξιολόγησης επίδοσης

Για την καλύτερη κατανόηση της προσέγγισης της λύσης του προβλήματος πρέπει να γίνει αναφορά στις μετρικές επίδοσης που χαρακτηρίζουν αν ένα σχήμα ασφαλείας είναι αποδοτικό και κατάλληλο για VANETs. Αρχικά, γίνεται αναφορά στις μετρικές των κρυπτογραφικών υπολογισμών, μιας και τα οχήματα έχουν περιορισμένη υπολογιστική ισχύ και οι RSU έχουν μεγάλο φόρτο εργασίας. Επιπλέον, γίνεται αναφορά στα μεγέθη και το πλήθος των μηνυμάτων στο δίκτυο και στην ενέργεια που καταναλώνεται από τα οχήματα κατά την επικοινωνία.

4.4.1 Μετρικές κρυπτογραφικών υπολογισμών

4.4.1.1 Μήκος κλειδιών

Το μήκος κλειδιών είναι η πιο χαρακτηριστική μετρική που αξιολογεί τους αλγόριθμους κρυπτογράφησης. Το μήκος συνήθως μετριέται σε bits και χαρακτηρίζει το επίπεδο ασφαλείας του αλγόριθμου συγκρίνοντάς τον με το επίπεδο ασφαλείας συμμετρικών αλγορίθμων κρυπτογράφησης, συνήθως του AES. Το μήκος του κλειδιού εξαρτάται άμεσα από τον ίδιο τον αλγόριθμο που το ορίζει και δεν είναι αναγκαίο μεγαλύτερο κλειδί να συνεπάγεται μεγαλύτερη ασφάλεια. Στα σύγχρονα συστήματα περιορισμένων πόρων, όπως σε IoT συσκευές και ενσωματωμένα συστήματα, μικρό μήκος κλειδιού είναι επιθυμητό για λόγους υπολογιστικής απόδοσης και μείωσης του απαιτούμενου χώρου αποθήκευσης. Επιπλέον, όπως θα εξηγηθεί παρακάτω, μικρότερα μήκη κλειδιών συνεπάγονται μικρότερα κρυπτογραφημένα κείμενα και άρα καλύτερη αξιοποίηση του εύρους ζώνης.

4.4.1.2 Χρόνος κρυπτογράφησης/αποκρυπτογράφησης

Ο χρόνος κρυπτογράφησης είναι μία πολύ σημαντική μετρική για τα συστήματα ασφαλείας. Είναι εύκολα κατανοητό πως αφορά τον συνολικό χρόνο που χρειάζεται ο αλγόριθμος κρυπτογράφησης να μετατρέψει το αρχικό κείμενο (plain text) σε κρυπτογραφημένη μορφή (cipher text) και το αντίστροφο (αποκρυπτογράφηση). Οι χρόνοι συνδέονται άμεσα με τη φύση του αλγόριθμου, καθώς και με το μέγεθος του κλειδιού που χρησιμοποιείται.

4.4.1.3 Χρόνος παραγωγής κλειδιών

Ο χρόνος παραγωγής κλειδιών συνήθως αφορά τη δημιουργία ζεύγους κλειδιών ασύμμετρης κρυπτογράφησης και αφορά το συνολικό χρόνο που χρειάζεται ο αλγόριθμος να τα δημιουργήσει και να τα αρχικοποιήσει. Για παράδειγμα στην κρυπτογραφία Ελλειπτικών Καμπυλών ορίζεται ως ο συνολικός χρόνος επιλογής ενός μυστικού τυχαίου αριθμού που θα χρησιμοποιηθεί για τον βαθμωτό πολλαπλασιασμό της γεννήτριας, το οποίο θα δημιουργήσει το δημόσιο κλειδί. Συνήθως αυτή η διαδικασία στα VANETs γίνεται πολλές φορές, καθώς δεν θεωρείται ασφαλές οι χρήστες να χρησιμοποιούν το ίδιο στατικό ζεύγος κάθε φορά που επικοινωνούν. Στην περίπτωση του σχήματος 9, τα κλειδιά μεταδίδονται και ανανεώνονται με τη χρήση πιστοποιητικών. Ο χρόνος παραγωγής κλειδιών εξαρτάται άμεσα από το επιθυμητό μέγεθος και τον αλγόριθμο που χρησιμοποιείται.

4.4.1.4 Χρόνος υπογραφής/επαλήθευσης

Όλα τα σχήματα ασφαλείας σε VANETs που αναλύθηκαν στην εργασία χρησιμοποιούν την τεχνική των ψηφιακών υπογραφών, για να εξασφαλίσουν την αυθεντικότητα και την ακεραιότητα των δεδομένων. Για την υλοποίηση τους χρησιμοποιείται κρυπτογραφία δημοσίου κλειδιού και είναι σημαντικό για την αξιολόγηση ενός σχήματος ο χρόνος που χρειάζεται ένας χρήστης να υπογράψει το μήνυμα που στέλνει, καθώς και ο χρόνος του παραλήπτη να την επικυρώσει, αφού πολλές χιλιάδες μηνύματα μπορεί να υπογράφονται και να επικυρώνονται κατά την επικοινωνία.

4.4.1.5 Χρόνος παραγωγής πιστοποιητικού και εξαγωγής κλειδιού

Τα ψηφιακά πιστοποιητικά είναι, επίσης, ένα πολύ διαδεδομένο εργαλείο στα σχήματα ασφαλείας για την διασφάλιση της γνησιότητας των παραμέτρων και την ταυτοποίηση των χρηστών. Οι υπολογισμοί που απαιτούν τα ψηφιακά πιστοποιητικά διαφέρουν σημαντικά ανάλογα με την υλοποίησή τους. Σε κάποια σχήματα, όπως το ECQV, απαιτούνται κρυπτογραφικοί υπολογισμοί κατά την παραγωγή τους και κατά την ανάκτηση του ζεύγους κλειδιών από αυτά. Επομένως, σε σχήματα ασφαλείας στα VANETs, όπου νέα πιστοποιητικά και κλειδιά παράγονται σχετικά συχνά, είναι απαραίτητη η αξιολόγηση του χρόνου παραγωγής και εξαγωγής των κλειδιών τους. Άλλου τύπου πιστοποιητικά βασίζονται απλώς στην επικύρωση μιας ψηφιακής υπογραφής.

4.4.1.6 Χρόνος κωδικοποίησης/αποκωδικοποίησης

Μερικές από τις κρυπτογραφικές τεχνικές δημοσίου κλειδιού, όπως η ECC, απαιτούν για την ασύμμετρη κρυπτογράφηση (τύπου ElGamal για παράδειγμα) το μήνυμα να κωδικοποιηθεί ως ένα στοιχείο του συνόλου που χρησιμοποιούν για την αριθμητική τους. Για παράδειγμα, στην κρυπτογραφία με βάση τις Ελλειπτικές/Υπερ-ελλειπτικές καμπύλες στην κρυπτογράφηση ElGamal το μήνυμα πρέπει να έρθει σε μορφή στοιχείου της Ομάδας, δηλαδή ενός σημείου στην καμπύλη ή ενός Divisor (βλ. κεφάλαιο 2.2). Επομένως, απαιτείται διαδικασία κωδικοποίησης/αποκωδικοποίησης για την πραγματοποίηση της κρυπτογράφησης και της αποκρυπτογράφησης και επομένως ο χρόνος της είναι σημαντικός στα σχήματα ασφαλείας. Συνδέεται άμεσα με τον αλγόριθμο της κωδικοποίησης που χρησιμοποιείται.

4.4.2 Μέγεθος μηνυμάτων

Το μέγεθος των μηνυμάτων που παράγονται θεωρείται ως μία ιδιαίτερα σημαντική μετρική στα δίκτυα των VANETs και γενικότερα σε δίκτυα περιορισμένων δυνατοτήτων. Πολύ μεγάλα μηνύματα μπορεί να οδηγήσουν σε συμφόρηση του

δικτύου και περαιτέρω αύξηση του χρόνου επικοινωνίας. Σε αυτή την περίπτωση, τα μηνύματα μπορεί να απαιτείται να κατακερματιστούν σε πολλά μικρότερα μηνύματα και να κατακλύσουν το δίκτυο. Το μέγεθος μετριέται σε bytes και συνδέεται άμεσα με τις κρυπτογραφικές τεχνικές που χρησιμοποιούνται και με το μέγεθος κλειδιού.

4.4.3 Κατανάλωση ενέργειας

Η συγκεκριμένη μετρική χρησιμοποιείται για να δώσει αίσθηση στον μηχανικό του φόρτου που προσδίδουν οι κρυπτογραφικές διαδικασίες και η επεξεργασία των μηνυμάτων σε συστήματα περιορισμένων δυνατοτήτων, όπως είναι τα ενσωματωμένα συστήματα. Στα οχήματα συγκεκριμένα, μεγάλη κατανάλωση ενέργειας μπορεί να οδηγήσει σε εξάντληση πόρων και σε απώλεια επικοινωνίας. Η εργασία αξιολογεί τις κρυπτογραφικές τεχνικές και με βάση την ενέργεια που καταναλώνουν.

Τελικά, η εργασία αποσκοπεί στη διερεύνηση κρυπτογραφικών τεχνικών οι οποίες μπορούν να βελτιώσουν την επίδοση των σχημάτων ασφαλείας σε VANETs, διατηρώντας ένα επιθυμητό επίπεδο ασφαλείας. Παρακάτω, αναλύονται τα εργαλεία που χρησιμοποιήθηκαν για την επίτευξη αυτού του στόχου και ακολουθεί ο τρόπος υλοποίησης της προσομοίωσης και των κρυπτογραφικών τεχνικών.

Κεφάλαιο 5: Προτεινόμενη Λύση: Σχεδιασμός και Υλοποίηση

Εφόσον πλαισιώθηκε αναλυτικά το πρόβλημα που αντιμετωπίζει η εργασία, είναι σημαντικό να γίνει σύντομη αναφορά στα εργαλεία και στις μεθόδους που επιλέχθηκαν να χρησιμοποιηθούν για την κατασκευή της λύσης του, πριν γίνει η παρουσίασή της. Επομένως, αρχικά, γίνεται ανάλυση των εργαλείων και του λογισμικού που χρησιμοποιήθηκε, οι αλγόριθμοι καθώς και οι λόγοι για τους οποίους επιλέχθηκαν. Στη συνέχεια, παρουσιάζεται η υλοποίηση που ακολουθήθηκε με τη βοήθεια των εργαλείων και των αλγορίθμων που αναφέρθηκαν στην αρχή. Αρχικά, παρουσιάζεται η υλοποίηση των κρυπτογραφικών τεχνικών με χρήση AES, ECC και HECC, ενώ για την HECC genus 3 παρουσιάζεται και η υλοποίηση της αντίστοιχης βιβλιοθήκης. Στη συνέχεια, παρουσιάζεται η υλοποίηση του οδικού δικτύου στον προσομοιωτή, η κίνηση των οχημάτων και η επικοινωνία τους με βάση το ασφαλές σχήμα [28]. Τέλος, παρουσιάζεται η υλοποίηση του μοντέλου ενέργειας που χρησιμοποιήθηκε για την αξιολόγηση της κατανάλωσης στο δίκτυο.

5.1 Εργαλεία/Λογισμικό

Αρχικά, γίνεται η παρουσίαση των εργαλείων που χρησιμοποιήθηκαν για την κατασκευή του περιβάλλοντος προσομοίωσης. Γίνεται αναφορά στα εργαλεία προσομοίωσης δικτύου (NS-3), στα εργαλεία προσομοίωσης κίνησης (SUMO) και στα εργαλεία απεικόνισης του δικτύου με κίνησης (PyViz).

5.1.1 NS-3

Το NS-3, ή αλλιώς Network Simulator 3 [29], είναι εργαλείο ανοιχτού κώδικα που χρησιμοποιείται για την κατασκευή προσομοιώσεων δικτυακών πρωτοκόλλων και συστημάτων. Το λογισμικό NS-3 είναι κατασκευασμένο για λειτουργικά συστήματα Linux, είναι γραμμένο σε C++, ενώ υποστηρίζει και Python scripting και είναι διάδοχος του NS-2. Υποστηρίζει πολλούς διαφορετικούς τύπους δικτύων και πρωτοκόλλων, ενώ είναι πολύ χρήσιμο για την μοντελοποίηση Αυτό-Οργανούμενων Δικτύων Οχημάτων, αφού έχει υλοποιημένη την τεχνολογία WAVE.

Το NS-3 είναι παρέχει στους χρήστες του τη δυνατότητα της επέκτασής του για τις δικές τους ανάγκες. Οι χρήστες μπορούν να ορίσουν τα δικά τους μοντέλα, πρωτόκολλα και εφαρμογές και να τα ενσωματώσουν στις προσομοιώσεις τους, πράγμα που το καθιστά κατάλληλο για ερευνητικούς και αναπτυξιακούς σκοπούς. Το NS-3 παρέχει τη δυνατότητα στους χρήστες του να επικεντρωθούν στο κομμάτι της στοίβας που τους ενδιαφέρει και να παρατηρήσουν αναλυτικά συμπεριφορές, αφού παρέχει ήδη υλοποιημένα πολλά από τα πρωτόκολλα δρομολόγησης (IP, AODV), μεταφοράς (TCP, UDP) και εφαρμογής (HTTPS, FTP), όλα με βάση τα επίσημα

πρότυπα του IEEE. Επιπλέον, το NS-3 είναι προσομοιωτής διακριτού χρόνου, που η προσομοίωση ορίζεται ως μία ακολουθία διακριτών γεγονότων. Τα γεγονότα μπορούν να είναι προκαθορισμένα από τον προγραμματιστή ή και δυναμικά, δηλαδή να ορίζονται με βάση κάποιες παραμέτρους κατά τη διάρκεια της προσομοίωσης, πράγμα που είναι ιδιαίτερα χρήσιμο για την προσομοίωση πραγματικών σεναρίων δικτυακής επικοινωνίας. Επιπρόσθετα, παρέχει τη δυνατότητα να συνεργαστεί με διάφορα άλλα εργαλεία, όπως αυτά που θα παρουσιαστούν παρακάτω, για την πιο ολοκληρωμένη προσομοίωση του δικτύου.

Είναι εύκολο να συμπεράνει κανείς από τα παραπάνω πως το NS-3 είναι ένα πολύ ισχυρό εργαλείο προσομοίωσης που μπορεί να βοηθήσει στην προσομοίωση του δικτύου ενδιαφέροντος της εργασίας. Ωστόσο, πέρα από αυτά, το εργαλείο NS-3 επιλέχθηκε για την κατασκευή της προσομοίωσης, καθώς παρέχει υλοποιημένη τη στοίβα πρωτοκόλλων του WAVE, δηλαδή το πρότυπο WiFi 802.11p καθώς και τα πρότυπα IEEE 1609, και δίνει και δυνατότητα παραμετροποίησης του (για παράδειγμα ορισμός του QoS), ενώ παρέχει υλοποιημένο και το πρωτόκολλο δρομολόγησης στα VANETs, το AODV. Εκτός αυτού, το NS-3 παρέχει μεθόδους για την μέτρηση της κατανάλωσης ενέργειας των συσκευών που απαρτίζουν την προσομοίωση και συγκεκριμένα της κατανάλωσης καρτών WiFi, που απαρτίζουν το WAVE. Χρησιμοποιήθηκε η έκδοση NS – 3.30.

5.1.2 SUMO (Simulation of Urban Mobility)

Το SUMO είναι και αυτό ένα εργαλείο προσομοίωσης, αλλά εξειδικεύεται στην προσομοίωση ρεαλιστικής κίνησης σε αστικά οδικά δίκτυα και δίνει τη δυνατότητα στους χρήστες να αναλύσουν και να βελτιστοποιήσουν τα οδικά δίκτυα. Εφαρμόζεται σε διάφορα επιστημονικά πεδία, όπως σε μηχανικούς μεταφορών, αρχιτέκτονες πόλεων και ερευνητές. Το SUMO [30] είναι ανοιχτού κώδικα λογισμικό γραμμένο σε C++, ενώ είναι διαθέσιμο για χρήση σε λειτουργικά Linux και Windows., ενώ ενημερώνεται διαρκώς σχετικά με τις νεότερες εξελίξεις στον τομέα των οδικών δικτύων.

Το SUMO παρέχει μία πληθώρα δυνατοτήτων για τη σχεδίαση σεναρίων αστικής κίνησης. Ο χρήστης μπορεί με λεπτομέρεια να μοντελοποιήσει δίκτυα, οχήματα, φανάρια, πεζούς, λεωφορεία, συστήματα ελέγχου της κυκλοφορίας και άλλα. Το SUMO μπορεί να μοντελοποιήσει τη συμπεριφορά των οχημάτων ανάλογα με διάφορους κανονισμούς οδικής κυκλοφορίας που μπορούν να οριστούν και δυναμικά από τον χρήστη, ενώ έχει τη δυνατότητα για εκτέλεση προσομοιώσεων με πολλές χιλιάδες οχήματα ανά δευτερόλεπτο.

Για την υλοποίηση της προσομοίωσης του οδικού δικτύου, το εργαλείο χρησιμοποιεί μικροσκοπική στρατηγική για τα οχήματα. Δηλαδή, κάθε όχημα έχει μία προκαθορισμένη διαδρομή η οποία ορίζεται από το σημείο εισόδου και εξόδου από την προσομοίωση και με αλγοριθμικό τρόπο καθορίζεται η πορεία του. Οι

αλγόριθμοι που χρησιμοποιούνται μπορούν να οριστούν από τον χρήστη, ενώ οι πιο διαδεδομένοι είναι οι Dijkstra και A*. Επιπλέον, ο χρήστης μπορεί να παραμετροποιήσει τον ρυθμό και την κατανομή άφιξης οχημάτων στην προσομοίωση και τον μέσο χρόνο που παραμένουν στο σύστημα. Το SUMO παρέχει διάφορες τοπολογίες προς χρήση στην προσομοίωση (grid, spider, random) και δίνει τη δυνατότητα στον χρήστη να εισαγάγει πραγματικούς δρόμους χρησιμοποιώντας χάρτες.

Επιπλέον, το SUMO, καθιστά εύκολη την εξαγωγή αρχείων XML για τη χρήση της προσομοίωσης σε άλλα εργαλεία για την περαιτέρω διερεύνηση του οδικού δικτύου. Αυτό που είναι ιδιαίτερα χρήσιμο στην παρούσα εργασία είναι πως παρέχει τη δυνατότητα εξαγωγής της προσομοίωσης σε αρχείο που μπορεί να ενσωματωθεί στο NS-2 και να δημιουργήσει κόμβους με κίνηση με πολύ γρήγορο τρόπο. Η ενσωμάτωση είναι δυνατή και στο NS-3, καθώς είναι συμβατό προς τα πίσω με διάφορες μεθόδους του NS-2.

5.1.3 PyViz

Το Python Visualizer, ή εν συντομία PyViz, είναι ένα εργαλείο απεικόνισης που είναι ενσωματωμένο στην διανομή του NS-3. Παρέχει έναν εύκολο και διαδραστικό τρόπο απεικόνισης των αποτελεσμάτων της προσομοίωσης και της συμπεριφοράς του δικτύου. Το PyViz αξιοποιεί διάφορες βιβλιοθήκες της Python για την δημιουργία της απεικόνισης και έχει τη δυνατότητα να αποτυπώσει αναλυτικά την δικτυακή τοπολογία, να εμφανίσει δικτυακές πληροφορίες σε κάθε κόμβο, να απεικονίσει την μεταφορά των πακέτων κ.α. Επιπλέον, επιτρέπει στον χρήστη να διακόπτει και να ξεκινάει ξανά την προσομοίωση σε οποιοδήποτε σημείο της και να την επιταχύνει ή να την επιβραδύνει. Είναι χρήσιμο για δίκτυα VANETs, ώστε να φαίνεται η κίνηση και η συμπεριφορά των οχημάτων σε πραγματικό χρόνο και να παρουσιάζονται τα μηνύματα που ανταλλάσσονται.

5.2 Αλγόριθμοι κρυπτογράφησης

Παρουσιάζονται οι αλγόριθμοι κρυπτογράφησης που επιλέχθηκαν να χρησιμοποιηθούν για την υλοποίηση του ασφαλούς σχήματος στην προσομοίωση.

5.2.1 ECC

Η κρυπτογραφία ελλειπτικών καμπυλών αναλύθηκε και ορίστηκε θεωρητικά στο 2^{ο} κεφάλαιο. Είναι ιδιαίτερα διαδεδομένη μορφή κρυπτογραφίας δημοσίου κλειδιού και εφαρμόζεται ιδιαίτερα σε συστήματα περιορισμένων πόρων, όπως είναι και τα VANETs. Λόγω των μαθηματικών ιδιοτήτων που εξηγήθηκαν προηγουμένως, οι ελλειπτικές καμπύλες παρέχουν ικανοποιητική ασφάλεια με σχετικά μικρό μέγεθος

κλειδιού, συγκριτικά με άλλους αλγόριθμους δημοσίου κλειδιού, όπως είναι ο RSA. Έχουν γίνει σημαντικές βελτιώσεις στην αριθμητική της κρυπτογραφικής μεθόδου με αποτέλεσμα να γίνει αποδοτική επεξεργαστικά, ενώ μειώνει και τον αποθηκευτικό χώρο που απαιτείται. Τα μικρότερα κλειδιά, επιπλέον, συνεπάγονται μικρότερα κρυπτογραφημένα μηνύματα, υπογραφές ή πιστοποιητικά και άρα το δίκτυο δεν κατακλύζεται από πολλά τμηματικά μηνύματα, λόγω κατακερματισμού. Παρακάτω γίνεται η σύγκριση των μεγεθών των κλειδιών του ECC και του RSA για το ίδιο επίπεδο ασφαλείας σε bits [31]:

Πίνακας 3: Σύγκριση μήκους κλειδιών ECC – RSA [31]

Symmetric	**RSA**	**ECC**
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

Η ECC βασίζεται στην δυσκολία του προβλήματος του διακριτού λογάριθμου και συγκεκριμένα το πρόβλημα στην ECC ονομάζεται ECDLP (Elliptic Curve Discrete Logarithm Problem) [4], [19], το οποίο εξηγήθηκε στο κεφάλαιο 2, όπως και ο λόγος που παρέχει καλύτερη ασφάλεια από τις άλλες Κυκλικές Ομάδες [11]. Η επιλογή της καμπύλης είναι ένα ιδιαίτερα σημαντικό ζήτημα για την ασφάλεια των συστημάτων βασισμένα στο ECDLP. Στην ECC έχει γίνει εκτενής έρευνα και επομένως παρέχονται έτοιμες ασφαλής καμπύλες και παράμετροι. Στα επόμενα υπό-κεφάλαια της υλοποίησης αναλύεται η επιλογή της καμπύλης στο σχήμα που βασίζεται η προσομοίωση.

Πληθώρα διαφορετικών εφαρμογών έχουν αναπτυχθεί βασισμένες στην ECC. Η ECC δίνει λύσεις για εφαρμογές ανταλλαγής κλειδιού με το ECDH (Elliptic Curve Diffie-Hellman), ασύμμετρης κρυπτογράφησης με την EC ElGamal κρυπτογράφηση κειμένου, ψηφιακών υπογραφών με τις υπογραφές EC ElGamal και ECDSA, οι οποίες είναι βασισμένες στις πρώτες, ψηφιακά πιστοποιητικά με το σχήμα ECQV. Επομένως, η ECC δίνει δυνατότητα για την κάλυψη όλων των απαιτούμενων κρυπτογραφικών εφαρμογών του ασφαλούς σχήματος [28]. Επομένως, έχει ιδιαίτερη ερευνητική σημασία η εφαρμογή ECC στο σχήμα για τη λύση του προβλήματος που προαναφέρθηκε, μιας και τα μεγέθη των κλειδιών είναι σημαντικά μικρότερα από τον RSA. Η υλοποίηση των παραπάνω εφαρμογών αναλύεται στα επόμενα υπό-κεφάλαια.

5.2.2 HECC

Οι επόμενοι αλγόριθμοι που απασχολούν την εργασία αφορούν την κρυπτογραφία Υπερελλειπτικών Καμπυλών, ή εν συντομία HECC, η θεωρία της οποία αναλύθηκε στο 2ο κεφάλαιο. Η HECC δεν είναι όσο διαδεδομένη είναι η ECC, καθώς συνήθως η ECC κάλυπτε τις ανάγκες σε ασφάλεια και μικρά κλειδιά. Ωστόσο, οι εξελίξεις στον τομέα του IoT επαναφέρουν την HECC στην επιφάνεια, αφού ερευνητικά έχει δείξει πως μπορεί να ανταγωνιστεί την ECC, αφού μεν η αριθμητική του είναι πιο δύσκολη, υπάρχει τρόπος να γίνει περισσότερο γρήγορη από αυτή της ECC, λόγω του μειωμένου μήκους κλειδιού που προσφέρει. Οι καμπύλες που μπορούν να έχουν εφαρμογή, ωστόσο, είναι οι καμπύλες γένους 2 και 3, καθώς σε μεγαλύτερη γένη το σχήμα δεν είναι επαρκώς ασφαλές [12]. Επομένως, στην εργασία επιλέγεται να χρησιμοποιηθεί HECC genus 2 και 3. Παρακάτω παρατίθεται ο πίνακας σύγκρισης μήκους κλειδιών σε bits του ECC και των HECC genus 2 και 3 με βάση τη θεωρία του κεφαλαίου 2 [12].

Πίνακας 4: Σύγκριση μήκους κλειδιών ECC – HECC [12]

Symmetric	ECC	HECC genus 2	HECC genus 3
80	160	80	54
112	224	112	75
128	256	128	86
192	384	192	128
256	512	256	171

Αντίστοιχα με προηγουμένως η HECC βασίζεται στη δυσκολία του προβλήματος HECDLP (Hyperelliptic Curve Discrete Logarithm Problem) και το πως αυτό χτίζεται με τη χρήση των Reduced Divisors (βλ. κεφάλαιο 2). Η επιλογή της καμπύλης είναι και εδώ μία πολύ σημαντική διαδικασία και αν δεν γίνει σωστά, μπορεί να δημιουργήσει κενά ασφαλείας. Η HECC δεν έχει λάβει τόση αναγνώριση όσο η ECC, επομένως δεν έχει γίνει τόση έρευνα για την παραγωγή ασφαλών και γρήγορων καμπυλών και την αποθήκευσή τους σε κάποια βάση δεδομένων. Επομένως, η επιλογή της καμπύλης γίνεται ένα πιο σύνθετο πρόβλημα. Στα επόμενα υπό-κεφάλαια, γίνεται ανάλυση των καμπυλών που χρησιμοποιήθηκαν και οι λόγοι επιλογής των παραμέτρων.

Η HECC μπορεί να χρησιμοποιηθεί στις ίδιες ακριβώς εφαρμογές που αναφέρθηκαν στην ECC, αφού στηρίζονται στο πρόβλημα του DLP, στο οποίο βασίζεται και η HECC. Επομένως, για τις ίδιες επιλογές σχημάτων κρυπτογράφησης, ανταλλαγής, υπογραφών, πιστοποιητικών μπορεί να χρησιμοποιηθεί αντίστοιχα η HECC. Ωστόσο, δεν υπάρχουν πολλές έτοιμες πηγές υλοποιημένων αλγορίθμων σε HECC, επομένως ήταν αναγκαία η υλοποίησή τους από την αρχή. Τα παραπάνω καθιστούν την HECC μία τεχνική με πολύ ερευνητικό ενδιαφέρον στα VANETs. Είναι σημαντικό η κοινότητα να μελετήσει αν η χρήση τους μπορεί να βελτιώσει τα σχήματα ασφαλείας στα VANETs. Τα μικρότερα κλειδιά, καθώς και η δυνατότητα η

αριθμητική της να γίνει γρηγορότερη από αυτή της ECC παρότρυναν τη χρήση τους στα σχήματα ασφαλείας των VANETs και επομένως επιλέχθηκαν στην παρούσα εργασία. Στα επόμενα υπό-κεφάλαια αναλύεται η υλοποίηση των κομματιών της HECC που δεν ήταν ήδη υλοποιημένα σε κάποια βιβλιοθήκη, καθώς και οι αντίστοιχες εφαρμογές HEC ElGamal encryption, signatures και HECQV πιστοποιητικών.

5.2.3 AES

Ο αλγόριθμος Advanced Encryption Standard, ή AES [32], είναι ο πιο διαδεδομένος αλγόριθμος συμμετρικής κρυπτογράφησης. Ο AES επεξεργάζεται την είσοδο ως μία σειρά από τμήματα (blocks), τα οποία έχουν συγκεκριμένο μέγεθος. Ο αλγόριθμος χρησιμοποιεί έναν συνδυασμό από προσθέσεις, αφαιρέσεις, μεταθέσεις και γραμμικούς μετασχηματισμούς για να κρυπτογραφήσει τα δεδομένα. Η αποκρυπτογράφηση ακολουθεί την αντίστροφη διαδικασία.

Η διαδικασία της κρυπτογράφησης και αποκρυπτογράφησης γίνεται με τη χρήση ενός μόνο συμμετρικού κλειδιού, το οποίο στον AES μπορεί να λάβει μεγέθη 128, 192 και 256 bits. Μεγαλύτερο μέγεθος κλειδιού συνεπάγεται μεγαλύτερη ασφάλεια. Ο αλγόριθμος αποτελείται από έναν αριθμό γύρων, ο οποίος εξαρτάται άμεσα από το μέγεθος του κλειδιού. Για 128-bit κλειδί, ο AES έχει 10 γύρους, για 192-bit κλειδί 12 και για 256-bit κλειδί 14. Κάθε γύρος αποτελείται από 4 στάδια:

- Στάδιο AddRoundKey: Ένα κλειδί 128-bit που έχει παραχθεί από το αρχικό κλειδί εφαρμόζεται στο block εισόδου με τη χρήση bitwise XOR.
- Στάδιο SubBytes: Αφαιρεί μη γραμμικά κάθε byte του block εισόδου με έναν προκαθορισμένο πίνακα αναφοράς που ονομάζεται S-box.
- Στάδιο ShiftRows: Μεταθέτει κυκλικά τα bytes κάθε γραμμής του block με τρόπο που εξαρτάται από τον αριθμό της γραμμής.
- Στάδιο MixColumns: Πραγματοποιεί αναστρέψιμο γραμμικό μετασχηματισμό για να αναμείξει τις στήλες του block εισόδου.

Ο αλγόριθμος είναι ανθεκτικός σε οποιαδήποτε επίθεση έχει σχεδιαστεί εναντίον του και χρησιμοποιείται πλέον σε πάρα πολλές εφαρμογές. Πολλά δικτυακά πρωτόκολλα βασίζονται σε αυτόν, όπως το IPSec, το TLS, το WiFi 802.11i, το SSH κ.α. Επιπλέον, είναι πάρα πολύ αποδοτικός σε επίπεδο λογισμικού και υλικού στα σημερινά υπολογιστικά συστήματα. Επομένως, εφόσον το σχήμα [28] που αναλύει η εργασία καταλήγει σε στάδιο επικοινωνίας με συμμετρική κρυπτογράφηση, η επιλογή τους AES για αυτή τη διαδικασία είναι ιδανική. Ωστόσο, δεν γίνεται κάποιου είδους αξιολόγησης του AES στην παρούσα εργασία.

5.3 Υλοποίηση κρυπτογραφικών τεχνικών

Η διαδικασία της επικοινωνίας και εγγραφής των οχημάτων και της RSU που προτείνεται στο σχήμα [28] απαιτεί διάφορες κρυπτογραφικές εφαρμογές, για να είναι ασφαλής. Οι επιλογές που έγιναν για τους αλγόριθμους κρυπτογράφησης, οι οποίοι αναλύθηκαν στα προηγούμενα υπό-κεφάλαια, καλύπτουν τις ανάγκες του σχήματος με τη χρήση των μεθόδων τους για κρυπτογράφηση, υπογραφές και πιστοποιητικά. Η συγκεκριμένη εργασία δεν επικεντρώνεται στη διαδικασία ανταλλαγής και ανανέωσης ψευδωνύμων και στη διαδικασία ανανέωσης και ανάκλησης κλειδιών. Η προσομοίωση σχεδιάστηκε με επίκεντρο τη διαδικασία της ένταξης του οχήματος στην περιοχή της RSU, στην αποδοχή του, στην εκλογή Group Leader και στην ενημέρωση του Group Leader ή της RSU για κάποια γεγονότα. Επομένως, είναι απαραίτητο να υλοποιηθούν:

- Παραγωγή κλειδιών συμμετρικών και ασύμμετρων.
- Συμμετρική και ασύμμετρη κρυπτογράφηση.
- Παραγωγή και επαλήθευση ψηφιακών υπογραφών.
- Παραγωγή πιστοποιητικών και εξαγωγή κλειδιών από αυτά.

Η συμμετρική κρυπτογράφηση υλοποιήθηκε με τη χρήση του AES, όπως προαναφέρθηκε. Η ασύμμετρη κρυπτογράφηση με ECC, HECC έγινε με τη χρήση της μεθόδου ElGamal [33]. Ωστόσο, η συγκεκριμένη μέθοδος απαιτεί από τα μηνύματα να είναι σε μορφή στοιχείου της κρυπτογραφικής Ομάδας, δηλαδή σημείου ή divisor. Επομένως, ήταν απαραίτητη η υλοποίηση αντιστρέψιμων συναρτήσεων για την αντιστοίχιση κειμένου σε σημείο/divisor. Στη συνέχεια, οι ψηφιακές υπογραφές υλοποιήθηκαν με την μέθοδο ECDSA [34] και με ElGamal signatures για HECC [33], καθώς η μέθοδος του ECDSA ουσιαστικά είναι μία άλλη μορφή της υπογραφής ElGamal. Η παραγωγή πιστοποιητικών και η εξαγωγή των κλειδιών έγινε με τη χρήση του ECQV [35], το οποίο επεκτάθηκε στην οικογένεια των Υπερελλειπτικών Καμπυλών (HECQV). Επιπλέον, δεν παρέχεται κάποια βιβλιοθήκη ανοιχτού κώδικα για HECC genus 3, επομένως ήταν απαραίτητο να υλοποιηθεί. Γίνεται αναφορά, επιπλέον, στην επιλογή των παραμέτρων των καμπυλών.

5.3.1 AES

Όπως προαναφέρθηκε, ο αλγόριθμος AES χρησιμοποιήθηκε σε όλες τις προσομοιώσεις για να καλύψει την ανάγκη της συμμετρικής κρυπτογράφησης στο ασφαλές σχήμα [28]. Χρησιμοποιείται σε συνδυασμό και με τις 3 περιπτώσεις αλγορίθμων ασύμμετρης κρυπτογράφησης (ECC, HECC genus 2, HECC genus 3). Για την υλοποίηση της κρυπτογράφησης και της παραγωγής κλειδιών και του IV χρησιμοποιήθηκε η βιβλιοθήκη Crypto++ της C++ με τις εξής επιλογές:

- Μέθοδος λειτουργίας: Cipher Block Chaining (CBC), καθώς θεωρείται πως παρέχει μεγαλύτερη ασφάλεια.
- Μήκος κλειδιού: 16 bytes – 128 bits για μεγαλύτερη ταχύτητα.
- Μήκος block: 128 bits.

Στο σχήμα [28], στους αλγόριθμους 1 και 3, η RSU ή ο GL πρέπει να παράγουν ένα συμμετρικό κλειδί και να το στείλουν στο όχημα που θέλει να κάνει Join στην περιοχή. Μαζί με το κλειδί στέλνεται και ο αντίστοιχος IV. Η μέθοδος παραγωγής αυτών των δύο γίνεται ως εξής [36]:

```
AutoSeededRandomPool prng;

SecByteBlock key(AES::DEFAULT_KEYLENGTH);
SecByteBlock iv(AES::BLOCKSIZE);

prng.GenerateBlock(key, key.size());
prng.GenerateBlock(iv, iv.size());

std::string keystr, ivstr;
HexEncoder encoder(new StringSink(keystr));
encoder.Put(key, key.size());
encoder.MessageEnd();

HexEncoder encoder2(new StringSink(ivstr));
encoder2.Put(iv, iv.size());
encoder2.MessageEnd();
```

Αρχικά παράγονται 2 byte blocks με τυχαία bytes με τη χρήση της συνάρτησης παραγωγής τυχαίων αριθμών AutoSeededRandomPool. Στη συνέχεια το κλειδί και ο IV μετατρέπονται σε Strings με τη χρήση του HexEncoder της Crypto++ που μετατρέπει Byte Blocks σε Strings.

Στη συνέχεια, παρουσιάζεται η διαδικασία κρυπτογράφησης/αποκρυπτογράφησης ενός μηνύματος με AES. Για αυτό το σκοπό υλοποιήθηκαν δύο συναρτήσεις encrypt_message_AES και decrypt_message_AES [36]:

```
CBC_Mode<AES>::Encryption e;
```

```
    e.SetKeyWithIV(key, 16, iv);

    StreamTransformationFilter encfilter(e, nullptr,
BlockPaddingSchemeDef::PKCS_PADDING);
    encfilter.Put(in, size);
    encfilter.MessageEnd();
    encfilter.Get(out, size+16-size%16);
```

Η συνάρτηση δέχεται ως είσοδο δύο buffers που αποτελούνται από bytes, έναν εξόδου (out) και έναν εισόδου (in), το μέγεθος του buffer εισόδου και το κλειδί και τον IV σε string μορφή. Αφού γίνει η μετατροπή του κλειδιού και του IV σε bytes ξανά, τίθεται η μέθοδος της κρυπτογράφησης ως AES με CBC, δίνεται η είσοδος προς κρυπτογράφηση σε ένα StreamTransformationFilter, αφού η είσοδος είναι σε μορφή byte buffer και στη συνέχεια η έξοδος αποθηκεύεται στον buffer out. Ωστόσο, το μέγεθος της εξόδου ενδέχεται να είναι μεγαλύτερο αν το μέγεθος της εισόδου δεν είναι πολλαπλάσιο των 16 bytes, καθώς χρησιμοποιείται η τεχνική του PKCS Padding για να μετατρέψει το μέγεθος της εισόδου σε ακέραιο πολλαπλάσιο του μεγέθους του block.

Στην αποκρυπτογράφηση ακολουθείται η αντίστροφη διαδικασία. Η μετατροπή του κλειδιού και του IV σε Bytes είναι ίδια οπότε παρατίθεται μόνο η διαδικασία αποκρυπτογράφησης:

```
        CBC_Mode<AES>::Decryption d;
        d.SetKeyWithIV(key, 16, iv);

        StreamTransformationFilter decfilter(d, nullptr,
BlockPaddingSchemeDef::PKCS_PADDING);
        decfilter.Put(in, size);
        decfilter.MessageEnd();
        decfilter.Get(out, size);
```

Τα ορίσματα της συνάρτησης decrypt_message_AES είναι ίδια, απλώς έξοδος είναι πλέον το αποκρυπτογραφημένο μήνυμα. Σε περίπτωση κάποιου λάθους στην αποκρυπτογράφηση (όταν δηλαδή το κλειδί ή ο IV είναι λάθος) η μέθοδος decfilter.Put πετάει Exception, το οποίο πιάνεται με try, catch και τυπώνεται.

5.3.2 ECC

Για την υλοποίηση των κρυπτογραφικών εφαρμογών με ECC χρησιμοποιήθηκε ξανά η βιβλιοθήκη Crypto++, η οποία παρέχει πολλές από αυτές υλοποιημένες. Ωστόσο, η κρυπτογράφηση ElGamal δεν υπάρχει υλοποιημένη και υλοποιήθηκε με το API που παρέχει η Crypto++ για πράξεις, ενώ υλοποιήθηκε η

μέθοδος Koblitz για τα Encodings των μηνυμάτων στην Καμπύλη. Επίσης, υλοποιήθηκε το σχήμα ECQV για τα πιστοποιητικά και χρησιμοποιήθηκαν υπογραφές ECDSA. Η καμπύλη που επιλέχθηκε είναι μία από τις πιο διαδεδομένες ασφαλείς και γρήγορες καμπύλες, η secp256r1 [37], η οποία παράγει κλειδιά μεγέθους 256 bits και συνεπώς επίπεδο ασφαλείας 128 bits.

5.3.2.1 Παραγωγή ζεύγους κλειδιών

Η παραγωγή του ζεύγους κλειδιών αποτελείται από 2 απλά βήματα. Παράγεται ένας τυχαίος ακέραιος αριθμός με τιμές από το 1 έως την τάξη της Ομάδας (256 bits κοντά στο μέγεθος της παραμέτρου p που χρησιμοποιείται) και στη συνέχεια πολλαπλασιάζεται βαθμωτά η γεννήτρια G για την παραγωγή του δημοσίου κλειδιού. Επομένως:

```
CryptoPP::Integer x(prng, CryptoPP::Integer::One(), veh1ec-
>group.GetMaxExponent());
Element h = veh1ec->group.ExponentiateBase(x);
```

Η Crypto++ παρέχει constructor ενός αριθμού με βάση μια γεννήτρια τυχαίων αριθμών (prng) και στη συνέχεια με τη χρήση της μεθόδου ExponentiateBase πολλαπλασιάζεται βαθμωτά η γεννήτρια της ομάδας με το μυστικό κλειδί και παράγεται το δημόσιο κλειδί h [38].

5.3.2.2 Κρυπτογράφηση / Αποκρυπτογράφηση ElGamal

Η κρυπτογράφηση ElGamal [33] είναι πρακτικά δύο πράξεις σημείων της Ελλειπτικής:

$a = k * G,\ \ b = k * P + M$, όπου το παραγόμενο κρυπτογραφημένο κείμενο είναι η τούπλα (a, b). Το k είναι ένας τυχαίος αριθμός από το 1 έως την τάξης της Ομάδας (256 bits). Το G είναι το σημείο γεννήτριας, το P είναι το δημόσιο κλειδί και το M είναι το μήνυμα κωδικοποιημένο ως σημείο της καμπύλης. Επομένως:

```
a = veh1ec->group.ExponentiateBase(k);
btemp = veh1ec->group.GetCurve().ScalarMultiply(veh1ec->rsupub,
k);
b = veh1ec->group.GetCurve().Add(btemp, m);
```

Η αποκρυπτογράφηση αντίστοιχα ορίζεται από την πράξη $M = b - x * a$, όπου x είναι το μυστικό κλειδί του παραλήπτη. Επομένως:

```
mtemp = group.GetCurve().ScalarMultiply(a, rsu1ec->priv);
m = group.GetCurve().Subtract(b, mtemp);
```

5.3.2.3 Encoding / Decoding

Η ανάγκη για την κωδικοποίηση και αποκωδικοποίηση του κειμένου ως σημείο στην καμπύλη οδήγησε στην αναζήτηση και υλοποίηση αντίστοιχων αλγορίθμων. Χρησιμοποιήθηκε η κωδικοποίηση Koblitz (Koblitz encodings [39]), η οποία λειτουργεί ως εξής: Αν το μήνυμα που πρέπει να κωδικοποιηθεί είναι ένας ακέραιος αριθμός x, τότε το σημείο της καμπύλης P που θα κωδικοποιηθεί έχει ως συντεταγμένη x επί κάποιον ακέραιο k (συνήθως 100 ή 1000) τον ίδιο τον ακέραιο και ως y τη λύση της εξίσωσης της καμπύλης για το δεδομένο x. Αν αυτό δεν λύνεται, τότε το x αυξάνεται κατά 1 και ακολουθείται η ίδια διαδικασία το πολύ k φορές. Αν φτάσει ο αλγόριθμος στο k σημαίνει πως αποτυγχάνει η κωδικοποιήσει. Όσο μεγαλύτερο είναι το k, τόσο πιο απίθανο αυτό να συμβεί. Επιπλέον, για να υπολογιστεί το y από την εξίσωση της καμπύλης πρέπει να υπολογιστεί η τετραγωνική ρίζα στην έννοια της αριθμητικής υπολοίπου (modular arithmetic). Επίσημα ονομάζεται quadratic residue και ορίζεται ως $y^2 = q \pmod n$. Με τη χρήση της βιβλιοθήκης NTL, η οποία παρέχει αριθμητική υπολοίπου με γρήγορες πράξεις, υλοποιείται η τεχνική Koblitz:

```
int try_koblitz (poly_t f, ZZ x, ZZ_p &x1, ZZ_p &y1, ZZ k, ZZ p)
{
  ZZ_p cand_y;
  ZZ_p yfound;
  for (int i=0; i < k; i++) {
    x1 = to_ZZ_p(x*k + i);
    cand_y = eval(f, x1);
    yfound = squareRoot(cand_y, p);
    if(yfound != 0) {
      y1 = yfound;
      return 0;
    }
  }
  return 1;
}
```

Η παράμετρος f αναφέρεται στο πολυώνυμο f της εξίσωσης της καμπύλης, το x είναι το μήνυμα εισόδου ως ακέραιος, τα x1, y1 είναι το σημείο εξόδου της κωδικοποίησης και το p είναι η παράμετρος του Πεπερασμένου Σώματος στο οποίο ορίζεται η καμπύλη. Η μέθοδος eval λύνει την εξίσωση f για είσοδο $x1 = x * k + i$ και παράγει το y^2. Υπολογίζεται η ρίζα του με τη χρήση της συνάρτησης squareRoot (που αναφέρεται στο quadratic residue [40]) και αν υπολογίζεται, τότε η συνάρτηση επιστρέφει αλλιώς επαναλαμβάνεται η διαδικασία. Το k επιλέγεται να είναι 1000. Η συνάρτηση

```
Element text_to_ecpoint(std::string txt, int len, GroupParameters
group, int size)
```

μετατρέπει το μήνυμα αρχικά σε έναν ακέραιο αριθμό με τη χρήση της NTL [41], εξάγει την συνάρτηση f από την καμπύλη με τη χρήση του API και τελικά καλεί την try_koblitz για την κωδικοποιήσει σε συντεταγμένες. Αν το μήνυμα είναι μεγαλύτερο από 256 bits, τότε χρειάζεται κατάτμηση και η συνάρτηση επιστρέφει Error. Μετά την κωδικοποίηση σε συντεταγμένες το σημείο κατασκευάζεται ως Element της καμπύλης:

```
CryptoPP::Integer xcpp, ycpp;
xcpp.Decode(xp, size);
ycpp.Decode(yp, size);
Element point(xcpp, ycpp);
return point;
```

Η μέθοδος Decode μετατρέπει σε ακέραιους τις συντεταγμένες από byte array και κατασκευάζει το σημείο.

Η αποκωδικοποίηση γίνεται με τη συνάρτηση ecpoint_to_text. Για να αποκωδικοποιηθεί το κείμενο πρέπει απλώς να διαιρεθεί ακέραια η συντεταγμένη x του σημείου με το k και να μετατραπεί ο ακέραιος σε char array:

```
    CryptoPP::Integer encmess = point.x/1000;
```

5.3.2.4 Υπογραφή / Επικύρωση

Για τις υπογραφές η εργασία χρησιμοποιεί τις υλοποιημένες υπογραφές ECDSA της βιβλιοθήκης Crypto++ [42]:

```
ECDSA<ECP, SHA256>::Signer signer(k1);
size_t siglen = signer.MaxSignatureLength();
std::string signature(siglen, 0x00);
siglen = signer.SignMessage( prng, message, size,
(byte*)&signature[0] );
```

Αρχικά, αρχικοποιείται ο αλγόριθμος ECDSA με το μυστικό κλειδί του υπογράφων και στη συνέχεια με τη χρήση της κλάσης Signer του ECDSA και με χρήση SHA256 για το hashing του μηνύματος παράγεται η υπογραφή σε μορφή string.

Η επικύρωση γίνεται ως εξής:

```
ECDSA<ECP, SHA256>::PublicKey publicKey;
publicKey.Initialize(ASN1::secp256r1(), Pk);
ECDSA<ECP, SHA256>::Verifier verifier(publicKey);
bool result = verifier.VerifyMessage( message, size, (const
byte*)&sig[0], sig.length());
```

Εδώ, αρχικοποιείται το δημόσιο κλειδί του αποστολέα για την επικύρωση και με τη χρήση της κλάσης Verifier επικυρώνεται η υπογραφή στο μήνυμα. Η μέθοδος VerifyMessage επιστρέφει true αν είναι επιτυχής, αλλιώς false.

5.3.2.5 Πιστοποιητικά ECQV

Με σκοπό τη διάδοση των κλειδιών μέσω των πιστοποιητικών από την CA, την ανανέωσή τους και την εξασφάλιση της γνησιότητας των κλειδιών χρησιμοποιείται το σχήμα ECQV [35]. Το σχήμα αποτελείται από 4 αλγόριθμους: ECQV_Setup, Cert_Request, Cert_Generate, Cert_PK_Extraction, Cert_Reception, οι οποίοι χρησιμοποιούν τις ιδιότητες της ECC. Η κωδικοποίηση των πιστοποιητικών για λόγους επίδοσης επιλέγεται να είναι απλώς με κάποια προκαθορισμένα πεδία συγκεκριμένου μεγέθους (fixed length fields). Συγκεκριμένα το πιστοποιητικό περιέχει ένα πεδίο name που εμπεριέχει το όνομα του χρήστη, το πεδίο issued_by που αναφέρεται το όνομα της CA που το εξέδωσε, το issued_on με την ημερομηνία έκδοσης, το expires_on με την ημερομηνία λήξης του και μετά ένα δημόσιο κλειδί από το οποίο εξάγεται το πραγματικό δημόσιο κλειδί του χρήστη με τη χρήση του δημοσίου κλειδιού της CA. Η επικοινωνία με την CA δεν είναι στα πλαίσια της εργασίας, επομένως, στην προσομοίωση τα πιστοποιητικά δημιουργούνται στους κόμβους με ένα προκαθορισμένο ζεύγος κλειδιών της CA.

Ως ιδιωτικό κλειδί της CA επιλέγεται ένας ακέραιος τυχαία από το 1 έως την τάξη της Ομάδας και το δημόσιο κλειδί είναι η γεννήτρια πολλαπλασιασμένη βαθμωτά με το ιδιωτικό κλειδί. Επομένως, ο αλγόριθμος ECQV_Setup παραλείπεται, εφόσον χρησιμοποιούνται αυτά τα κλειδιά. Στο Cert_Request ο χρήστης U που επιθυμεί ένα πιστοποιητικό παράγει ένα ζεύγος κλειδιών (k_U, R_U) με τον ίδιο τρόπο και στέλνει στην CA το δημόσιο κλειδί μαζί με την ταυτότητά του. Στο στάδιο Cert_Generate η CA παράγει ένα νέο ζεύγος κλειδιού (k, kG) και υπολογίζει το $P_U = R_U + kG$ και δημιουργεί το πιστοποιητικό. Παράλληλα, υπολογίζει το $r = e*k + ca\_priv \pmod n$, όπου $e = Hash(Cert_U)$, με μία συνάρτηση Hash από τους integer στους integer modulo n και ca_priv το μυστικό κλειδί της CA:

```
CryptoPP::Integer k1(prng, CryptoPP::Integer::One(),
group.GetMaxExponent());
Element kG = group.ExponentiateBase(k1);
Element pu1 = group.GetCurve().Add(ru, kG);
```

Παραπάνω παράγεται το P_U από την CA με βάση το R_U του χρήστη με χρήση των μεθόδων της Crypto++.

```
hash.Update(encoded, 31 + size + 1);
std::string digest;
digest.resize(hash.DigestSize());
hash.Final((byte*)&digest[0]);
```

Με τον παραπάνω κώδικα παράγεται το Hash του $Cert_U$. Επιλέγεται η συνάρτηση SHA3-256 για το Hashing που εξασφαλίζει επιπλέον ότι το αποτέλεσμα e θα είναι σχεδόν πάντα στο σύνολο των integer modulo n.

```
CryptoPP::ModularArithmetic mod(n);
this->r = mod.Multiply(hashed_p, k1);
this->r = mod.Add(this->r, capriv);
```

Παραπάνω υπολογίζεται το r με τη χρήση αριθμητικής modulo n που παρέχει η Crypto++. Ως hashed_p θεωρείται το e, δηλαδή το Hash($Cert_U$). Στη συνέχεια έχει παραχθεί το πιστοποιητικό ενώ έχουν περαστεί και τα ανάλογα πεδία.

Στο στάδιο Cert_PK_Extraction, οποιοσδήποτε που κατέχει το δημόσιο κλειδί της CA μπορεί να υπολογίσει το δημόσιο κλειδί Q_U του χρήστη U από το P_U. Η πράξη που εκτελείται είναι $Q_U = e*P_U + Q_{CA}$, όπου e = Hash($Cert_U$) και Q_{CA} είναι το δημόσιο κλειδί της CA:

```
this->qu = this->group.GetCurve().ScalarMultiply(this->pu,
hashed_p);
this->qu = this->group.GetCurve().Add(this->qu, this->capk);
```

Στο τελευταίο στάδιο Cert_Reception, μόνο ο χρήστης που έκανε το αίτημα για νέο πιστοποιητικό στη CA μπορεί να υπολογίσει το μυστικό κλειδί του από το r που έλαβε από την CA με τη χρήση του k_U μυστικού κλειδιού που έθεσε στο στάδιο Cert_Request. Αυτό γίνεται με την πράξη $d_U = r + e*k_U \pmod n$. Στη συνέχεια ελέγχεται αν $Q_U = d_U*G$, ώστε να επιβεβαιώσει ο χρήστης πως η εξαγωγές των κλειδιών έγιναν σωστά:

```
CryptoPP::ModularArithmetic mod(n);
CryptoPP::Integer du1 = mod.Multiply(hashed_p, ku);
du1 = mod.Add(du1, this->r);
Element qut = group.ExponentiateBase(du1);
if(group.GetCurve().Equal(this->qu, qut)) {
     this->du = du1;
     return 0;
}
```

5.3.3 HECC genus 2

Οι κρυπτογραφικές τεχνικές που επιλέχθηκαν για την ECC χρησιμοποιούνται και εδώ, με διαφορά τις υπογραφές οι οποίες δεν είναι HECDSA, αλλά υπογραφές ElGamal [33]. Εξάλλου οι υπογραφές HECDSA είναι μία παραλλαγή των υπογραφών ElGamal. Για την υλοποίησή τους χρησιμοποιήθηκε η βιβλιοθήκη libg2hec [43], η οποία παρέχει υλοποιημένες κλάσεις για καμπύλες, για divisors σε μορφή Mumford και υλοποιημένη αριθμητική με βάση τους αλγόριθμους του βιβλίου Handbook of Elliptic and Hyperelliptic Curve Cryptography των Henri Cohen και Gerhard Frey [44]. Η βιβλιοθήκη χρησιμοποιεί αριθμητική modulo p με τη χρήση της βιβλιοθήκης NTL (Number Theory Library) [41] και συγκεκριμένα της έκδοσης 5.5. Η NTL παρέχει μεθόδους για αριθμητική modulo p μεγάλων αριθμών και πολυωνύμων πράγμα που την καθιστά ιδιαίτερα χρήσιμη για τις κρυπτογραφικές τεχνικές. Επίσης, οι νεότερες εκδόσεις επιταχύνουν τη διαδικασία με τη χρήση της βιβλιοθήκης GMP. Η βιβλιοθήκη libg2hec είναι η βάση της υλοποίησης της αντίστοιχης βιβλιοθήκης για HECC genus 3 που αναφέρεται στο επόμενο υπό-κεφάλαιο.

5.3.3.1 Επιλογή καμπύλης

Όπως έχει προαναφερθεί στην εργασία, η επιλογή της καμπύλης είναι ένας πολύ σημαντικός παράγοντας της ασφάλειας και της επίδοσης της HECC. Ωστόσο, δυστυχώς, εδώ δεν υπάρχει η δυνατότητα από τη βιβλιοθήκη ή από κάποια βάση δεδομένων να επιλεχθεί εύκολα μία ασφαλής και αποδοτική καμπύλη. Η επιλογή της καμπύλης για όλες τις κρυπτογραφικές τεχνικές εκτός αυτής των υπογραφών γίνεται με βάση τις καμπύλες που ορίζονται από την μέθοδο των Encodings που θα εξηγηθεί παρακάτω. Επιλέγεται καμπύλη πάνω σε Πεπερασμένο Σώμα 128-bits, ώστε να δημιουργεί Ομάδα τάξης 256-bits και να υπάρχει σύγκριση με την ECC. Ωστόσο, για τις υπογραφές ElGamal είναι απαραίτητο να είναι γνωστή η τάξη της Ομάδας (Group Order) για να υλοποιηθούν με σωστό τρόπο. Από τη βιβλιογραφία, στη δημοσίευση "Construction of Secure Random Curves of Genus 2 over Prime Fields" [45] από τους Gaudry και Schost επιλέγεται μία καμπύλη με γνωστή τάξη Ομάδας που είναι πρώτος αριθμός 168-bits. Επομένως, το επίπεδο ασφαλείας των υπογραφών είναι 84 bits. Συγκεκριμένα η καμπύλη είναι η εξής:

$$y^2 = f(x)\ στο\ F_p\ με\ p = 5 \times 1024 + 8503491, \text{ με}$$

$$f(x) = x^5 + 268281082283935564490073 6x^3 + 226591355295993102902116x^2 + 254767471595292971789991 8x + 479730995970848967305935 0$$

Η τάξη της ομάδας που παράγει είναι:

$$N = 249999999999941304386009994022094639661975160756 9$$

όπου το N είναι πρώτος, πράγμα που είναι ιδανικό για τις υπογραφές.

Αρχικά, ορίζεται το Πεπερασμένο Σώμα όπου υλοποιείται η αριθμητική modulo p με βάση το p που προαναφέρθηκε με τη χρήση της μεθόδου field_t::init. Στη συνέχεια, ορίζεται το πολυώνυμο f με τους συντελεστές που προαναφέρθηκαν και τίθεται το πολυώνυμο στην καμπύλη με τη μέθοδο curve.set_f(). Στη συνέχεια, καλείται η μέθοδος curve.update(), η οποία ελέγχει αν η καμπύλη είναι έγκυρη με βάση τις προϋποθέσεις που αναφέρθηκαν στο κεφάλαιο 2.

5.3.3.2 Παραγωγή ζεύγους κλειδιών

Ξανά η παραγωγή του ζεύγους κλειδιών γίνεται με επιλογή ενός τυχαίου ακέραιου αριθμού από 1 έως τάξη Ομάδας, ο οποίος είναι και το μυστικό κλειδί και μετά πολλαπλασιάζεται βαθμωτά η βάση με αυτόν για να παραχθεί το δημόσιο κλειδί. Η γεννήτρια, εδώ, παράγεται με τυχαίο τρόπο και στην περίπτωση των υπογραφών είναι απαραίτητο να ελεγχθεί αν η γεννήτρια δημιουργεί μία Ομάδα τάξης ίδιας με το N. Επομένως:

```
RandomBnd(x, ptest*ptest);
```

Με αυτό τον τρόπο παράγεται τυχαία το μυστικό κλειδί στην περίπτωση που η τάξη δεν είναι γνωστή. Εφόσον είναι γνωστό πως η τάξη είναι κοντά στο p^2. Όταν είναι γνωστή η τάξη στην περίπτωση των υπογραφών:

```
do {
    RandomBnd(x, order);
} while (IsZero(x));
```

Στη συνέχεια, η παραγωγή του δημοσίου κλειδιού:

```
h = x * g;
```

Ο συντελεστής * αναφέρεται στον βαθμωτό πολλαπλασιασμό και το g είναι η γεννήτρια.

5.3.3.3 Κρυπτογράφηση / Αποκρυπτογράφηση ElGamal

Η κρυπτογράφηση και αποκρυπτογράφηση των μηνυμάτων με ασύμμετρο τρόπο γίνεται ξανά με κρυπτογράφηση ElGamal. Η πράξη που εκτελείται εξηγήθηκε αναλυτικά στο κεφάλαιο 5.3.2.2 για την ECC. Αντίστοιχα και εδώ η πράξη που ακολουθείται είναι ίδια. Επομένως:

```
NS_G2_NAMESPACE::divisor a, b;
RandomBnd(k, ptest*ptest);
a = k*g;
b = k*rsupub + m;
```

Το κρυπτογραφημένο κείμενο είναι η τούπλα (a, b). Η μεταβλητή rsupub αντικατοπτρίζει το δημόσιο κλειδί της RSU και m είναι το κείμενο προς κρυπτογράφηση κωδικοποιημένο σε μορφή divisor.

Η αποκρυπτογράφηση γίνεται ως εξής:

```
m = b - rsu1g2->priv*a;
```

Όπου η μεταβλητή rsu1g2->priv είναι το ιδιωτικό κλειδί της RSU.

5.3.3.4 Encoding / Decoding

Όπως και στην ECC, για την κρυπτογράφηση και αποκρυπτογράφηση ElGamal είναι απαραίτητο να υπάρχει μία διαδικασία που θα αντιστοιχίζει ένα μήνυμα, ή πρακτικά έναν αριθμό, στο στοιχείο της Ομάδας όπου εκτελείται η αριθμητική, δηλαδή εδώ σε Divisor. Έχουν υπάρξει πολλές προσπάθειες ανά τα χρόνια να δημιουργηθεί μια τέτοια διαδικασία. Το 2018, οι Michel Seck και Nafissatou Diarra δημιούργησαν έναν γενικευμένο αλγόριθμο για Encodings οποιουδήποτε γένους σε μία οικογένεια Υπερελλειπτικών Καμπυλών με κρυπτογραφικό ενδιαφέρον [46]. Ο αλγόριθμος αντιστοιχίζει ακέραιους σε σημεία στην Υπερελλειπτική Καμπύλη. Ωστόσο, έπειτα πρέπει να γίνει αντιστοίχιση των σημείων της καμπύλης σε Divisor.

Με βάση τη θεωρία (βλ. κεφάλαιο 2.2.4) ένας Divisor είναι πρακτικά ένα πολυώνυμο που παράγεται από τη x συντεταγμένη g σημείων, όπου g είναι το γένος. Για αρχή, ο κώδικας των Seck και Diarra μεταφράστηκε από τη γλώσσα Sage σε C++ για την αντιστοίχιση του μηνύματος σε σημεία στην καμπύλη και δημιουργήθηκε η κλάση Unified Encoding. Ο κώδικάς της παρατίθεται στο παράρτημα.

Input: The hyperelliptic curve $\mathbb{H}_g$, an element $r \in \mathcal{R}_g$
Output: A point (x, y) on $\mathbb{H}_g$
$v := v(g) = w[ur^2(-m_g s - n_g) - 1];$
$\varepsilon := \chi(v^{(2g+1)} + a_{(2g-1)}v^{(2g-1)} + a_{(2g-3)}v^{(2g-3)} + \ldots + a_1 v + a_0);$
$x := \frac{1+\varepsilon}{2}v + \frac{1-\varepsilon}{2}\left(\frac{w(-v+w)}{v+w}\right);$
$y := -\varepsilon\sqrt{x^{(2g+1)} + a_{(2g-1)}x^{(2g-1)} + a_{(2g-3)}x^{(2g-3)} + \ldots + a_1 x + a_0};$
return (x, y).

Εικόνα 19: Αλγόριθμος κωδικοποίησης σε Υπερελλειπτική Καμπύλη γένους g [46]

Τα Encodings που παράγονται από την κλάση Unified Encoding αφορούν μία συγκεκριμένη οικογένεια καμπυλών της μορφής:

$$H_g: \quad y^2 = x^{(2g+1)} + a_{(2g-1)} * x^{(2g-1)} + a_{(2g-3)} * x^{(2g-3)} + \cdots + a_1 * x + a_0$$

Οι συντελεστές a του πολυωνύμου έχουν επίσης συγκεκριμένη μορφή που εξαρτάται από το γένος της καμπύλης:

Πίνακας 5: Πίνακας παραμέτρων Καμπύλης κωδικοποίησης [46]

Genus g	a_1	$a_{(2g-1)}$	a_0	$a_3, g \geq 3$
1	$\frac{sw^2}{1}$	sw^2	$\frac{s-3}{3}w^3$	-
2	$a_1 = \frac{sw^4}{2}$	sw^2	$\frac{s-10}{10}w^5$	-
3	$a_1 = \frac{sw^6}{3}$	sw^2	$\frac{s-21}{21}w^7$	$\frac{5sw^4}{3}$
4	$a_1 = \frac{sw^8}{4}$	sw^2	$\frac{s-36}{36}w^9$	$\frac{7sw^6}{3}$
5	$a_1 = \frac{sw^{10}}{5}$	sw^2	$\frac{s-55}{55}w^{11}$	$3sw^8 = \frac{9}{3}sw^8$

Η παράμετρος w ορίζεται αυθαίρετα από τον χρήστη, ενώ η παράμετρος s κατασκευάζεται επίσης από μία αυθαίρετη παράμετρο που δίνει ο χρήστης, την παράμετρο u. Επομένως, η κλάση Unified Encoding, κατά την αρχικοποίησή της, ορίζει και την εξίσωση της καμπύλης που θα χρησιμοποιηθεί, με τη χρήση της libg2hec και έτσι, η καμπύλη που χρησιμοποιείται για τις εφαρμογές με HECC genus 2, εκτός των υπογραφών, παράγεται από την κλάση Unified Encoding και επιστρέφεται με τη μέθοδο getcurve().

```
UnifiedEncoding enc(ptest, 10, 4, 2);
curve = enc.getcurve();
```

Επίσης, το για το p που επιλέγεται πρέπει να ισχύει p = 7 mod 8 και p = 3 mod 4. Άρα για την HECC genus 2 επιλέγεται

$$p = 340282366920938463463374607431768211223$$

Τυχαία τέθηκαν u = 10 και w = 4.

Για την αντιστοίχιση 2 σημείων σε Divisor δημιουργήθηκαν οι συναρτήσεις points_to_divisor και divisor_to_points. Το πολυώνυμο u(x) του Divisor παράγεται ως $u(x) = (x - x_1) * (x - x_2)$, άρα $u(x) = x^2 + a * x + b$, όπου $a = -x_1 - x_2$ και $b = x_1 * x_2$ και το $v(x) = c * x + d$, όπου πρέπει να ισχύει $c * x_1 + d = y_1$ και $c * x_2 + d = y_2$, άρα $c = (y_1 - y_2)/(x_1 - x_2)$ και $d = y_1 - c * x_1$. Επομένως, από 2 έγκυρα σημεία δημιουργείται ένας έγκυρος divisor. Για την αντιστροφή, το μόνο που πρέπει να υπολογιστεί είναι οι ρίζες του $u(x)$ και από τις παραπάνω εξισώσεις υπολογίζονται ξανά τα y_1 και y_2 Οι συναρτήσεις points_to_divisor και divisor_to_points παρατίθενται στο παράρτημα.

Συνολικά, για να γίνει η αντιστοίχιση του μηνύματος σε divisor, η μέθοδος text_to_divisor κάνει στο μήνυμα padding, ώστε να φτιάξει ένα byte string το πολύ 30 bytes, το κόβει σε δύο ίσα μέρη των 15 bytes και στην αρχή προσθέτει ένα αναγνωριστικό byte με τιμή '1' ή '2', ώστε να διατηρηθεί στην αποκωδικοποίηση η σωστή σειρά των μερών του μηνύματος και μετατρέπει τα byte strings σε ακέραιο αριθμό. Στη συνέχεια μετατρέπει τα κομμάτια σε σημεία στην καμπύλη με την UnifiedEncoding::encode και με τα 2 σημεία παράγει τον divisor:

```
int fl = enc.encode(msgzz1, x1, y1);
if(fl) {
    std::cout << "Could not encode!" << std::endl;
    return 1;
}
fl = enc.encode(msgzz2, x2, y2);
if(fl){
    std::cout << "Could not encode!" << std::endl;
    return 1;
}
D = points_to_divisor(x1, y1, x2, y2, curve);
```

Η αποκωδικοποίηση κάνει την αντίστροφη διαδικασία. Ωστόσο, επειδή η μέθοδος UnifiedEncoding::decode επιστρέφει 2 τιμές, από τις οποίες μία μόνο είναι σωστή πρέπει να δοκιμαστούν τα αποτελέσματα για να βρεθεί το σωστό αρχικό κείμενο. Στη συγκεκριμένη περίπτωση η μέθοδος divisor_to_text ελέγχει αν από τα decoded σημεία προκύπτουν byte strings, όπου το πρώτο byte είναι '1' ή '2' και όλα τα υπόλοιπα bytes είναι ASCII.

```
divisor_to_points(D, x1, y1, x2, y2, p);
ZZ_p val1, val2, val3, val4;
int fl = enc.decode(val1, val2, x1, y1);
fl = enc.decode(val3, val4, x2, y2);
```

5.3.3.5 Υπογραφή / Επικύρωση

Για τις υπογραφές χρησιμοποιείται η καμπύλη που αναφέρθηκε στο κεφάλαιο 5.3.3.1. Οι υπογραφές ElGamal [33] υλοποιούνται ως εξής: Υπολογίζεται το $a = k * g$, όπου το g είναι η γεννήτρια και το k είναι ένας τυχαίος αριθμός από 0 έως το N, δηλαδή την τάξη της Ομάδας. Επιπλέον, υπολογίζεται το $b = (m - x * f(a))/k \ (mod\ N)$, όπου m είναι το Hash του μηνύματος προς υπογραφή, x είναι το μυστικό κλειδί του υπογράφοντος και f(a) είναι μία αντιστοίχιση ενός Divisor σε ακέραιο αριθμό. Η υπογραφή που παράγεται είναι η τούπλα (a, b). Η επικύρωση γίνεται με την πράξη $f(a) * h + b * a == m * g$, όπου h είναι το δημόσιο κλειδί.

Επομένως, πρέπει να παραχθεί αρχικά ένας divisor g που θα χρησιμοποιηθεί ως γεννήτρια και θα δημιουργεί Ομάδα τάξης N. Αυτό επιτυγχάνεται ως εξής:

```
do {
    g.random();
} while (g.is_unit() && !(g*order).is_unit());
```

Παράγεται ένας τυχαίος divisor έως ότου η πράξη g*N να δώσει μονάδα. Έτσι εξασφαλίζεται πως ο divisor είναι τάξης N.

Επιπλέον, η συνάρτηση που χρησιμοποιείται για την αντιστοίχιση του Divisor σε integer είναι υλοποιημένη από το παράδειγμα των υπογραφών της βιβλιοθήκης [43]:

```
static ZZ from_divisor_to_ZZ(const NS_G2_NAMESPACE::divisor& div,
const ZZ& n)
{
  poly_t u = div.get_upoly();
  ZZ temp = AddMod(sqr(rep(u.rep[0])), sqr(rep(u.rep[1])), n);
  return ( IsZero(temp) ? to_ZZ(1) : temp );
}
```

Δηλαδή, επιστρέφει τον ακέραιο που παράγεται από την πράξη $(u_1^2 + u_2^2)(modN)$, όπου u_1, u_2 είναι οι συντελεστές του πολυωνύμου u(x). Επομένως η υπογραφή υλοποιείται ως εξής:

```
do {
    RandomBnd(k, order);
} while (IsZero(k));
a = k * g;
f_a = from_divisor_to_ZZ(a, order);
/* b = (m - x*f(a))/k mod N */
b = ((m - x*f_a)*InvMod(k, order))%order;
```

Για το Hashing του μηνύματος χρησιμοποιείται ξανά η κλάση SHA3_224 της Crypto++.

Η επικύρωση υλοποιείται ως εξής:

```
ZZ f_a = from_divisor_to_ZZ(a, order);
if ( f_a * h + sigb * a == m * g )
```

Αν ισχύει η ισότητα, τότε η υπογραφή είναι έγκυρη.

5.3.3.6 Πιστοποιητικά HECQV genus 2

Το σχήμα ECQV [35] αναλύθηκε στο κεφάλαιο 5.3.2.5. Τώρα, για τα πιστοποιητικά με HECC genus 2 ακολουθείται η ίδια διαδικασία απλώς με στοιχεία της ομάδας να είναι οι Divisors. Η διαδικασία που ακολουθείται στο στάδιο Cert_Generate:

```
RandomBnd(k1, p*p);
NS_G2_NAMESPACE::divisor kG = k1*this->G;
NS_G2_NAMESPACE::divisor pu1 = ru + kG;
```

Με τον παραπάνω κώδικα παράγεται το P_U.

```
this->r = hashed*k1 + this->capriv;
```

Με τον παραπάνω κώδικα παράγεται το r. Να σημειωθεί πως όλες οι μεταβλητές είναι τύπου ZZ_p, δηλαδή οι πράξεις γίνονται modulo p.

Στο στάδιο Cert_PK_Extraction:

```
this->qu = hashed*this->pu + this->capk;
```

Με τον παραπάνω κώδικα υπολογίζεται το Q_U.

Στο στάδιο Cert_Reception:

```
ZZ du1 = (this->r + hashed*ku);
NS_G2_NAMESPACE::divisor qut = du1*this->G;
if(this->qu == qut) {
    this->du = du1;
    return 0;
}
```

Ο παραπάνω κώδικας υπολογίζει το d_U από το r και το k_U. Για το Hashing ξανά χρησιμοποιείται η κλάση SHA3_256 της Crypto++.

5.3.3.7 Divisor compression

Στην ECC, η βιβλιοθήκη Crypto++ δίνει τη δυνατότητα το δημόσιο κλειδί που περιγράφεται με ένα σημείο στην καμπύλη με συντεταγμένες x και y να συμπιεστεί, ώστε να μειωθεί το μέγεθός του από το μέγεθος του x συν το μέγεθος του y, δηλαδή συνολικά 2 φορές το μέγεθος της τάξης N, σε N bits συν ένα επιπλέον byte. Αυτό επιτυγχάνεται, καθώς το y είναι πρακτικά συνάρτηση του x και απαιτείται απλώς ένα αναγνωριστικό για να επιλεχθεί η σωστή λύση της εξίσωσης, καθώς υπολογίζεται η ρίζα του y που δίνει δύο λύσεις.

Στην HECC genus 2, οι Henri Cohen και Gerhard Frey στο βιβλίο τους [44] παραθέτουν έναν αλγόριθμο συμπίεσης των Divisor, ώστε να χρειάζεται να αποθηκευτούν μόνο οι συντελεστές του u(x) πολυωνύμου, αφού το v(x) μπορεί να προκύψει ως συνάρτηση του u(x). Συγκεκριμένα, ισχύει πως $u \mid v^2 - f$, δηλαδή πως το u διαιρεί ακριβώς το πολυώνυμο που προκύπτει από την πράξη $v^2 - f$. Από αυτό προκύπτει πως υπάρχει ένα πολυώνυμο s(x), τέτοιο ώστε $us = v^2 - f$, ή $us + f = v^2$. Η καμπύλη μπορεί να θεωρηθεί πως έχει τη μορφή:

$$y^2 = x^5 + f_4x^4 + f_3x^3 + f_2x^2 + f_1x + f_0$$

και επομένως με σύγκριση συντελεστών το s προκύπτει να έχει τη μορφή:

$s(x) = -x^3 + (u_1 - f_4)x^2 + (u_0 - u_1^2 + f_4u_1 - f_3)x + s_0$, όπου u_i είναι οι συντελεστές του πολυώνυμου u.

Τελικά προκύπτει πως:

$$\begin{aligned} u(x)s(x) + f(x) \\ &= \left(s_0 + f_2 - f_3u_1 - f_4(u_0 - u_1^2) + u_1(2u_0 - u_1^2)\right)x^2 \\ &+ \left(u_1s_0 + f_1 - f_3u_0 + f_4u_0u_1 + u_0(u_0 - u_1^2)\right)x + u_0s_0 + f_0 \quad (1) \end{aligned}$$

Η διακρίνουσα του παραπάνω πολυωνύμου είναι πρακτικά ένα πολυώνυμο ως προς s_0, εφόσον οι υπόλοιποι συντελεστές είναι γνωστοί. Επιπλέον, η διακρίνουσα είναι ίση με μηδέν καθώς το πολυώνυμο είναι ίσο με ένα τέλειο τετράγωνο $v^2(x)$. Προκύπτουν οι σχέσεις για τους συντελεστές του $v(x)$:

$$v_0^2 = u_0s_0 + f_0, \quad (2)$$

$$2v_0v_1 = u_1s_0 + f_1 - f_3u_0 + f_4u_0u_1 + u_0(u_0 - u_1^2), \quad (3)$$

$$v_1^2 = s_0 + f_2 + f_3u_1 - f_4(u_0 - u_1^2) + u_1(2u_0 - u_1^2) \quad (4)$$

Για τη συμπίεση, υπολογίζεται αρχικά το s_0 από την εξίσωση (2) αν το $u_0 \neq 0$, αλλιώς από την εξίσωση (4). Αν ισχύει πως $u_1^2 - 4u_0 \neq 0$, τότε αρκεί να λυθεί η διακρίνουσα της (1) και να βρεθεί ποια από τις 2 λύσεις δίνει τη σωστή τιμή s_0. Αυτή η απόφαση κωδικοποιείται ως 1 bit. Στην αριθμητική υπολοίπου, η λύση της διακρίνουσας θα δώσει 2 λύσεις, η μία μεγαλύτερη από την άλλη. Επομένως, αν η σωστή λύση είναι η μεγαλύτερη από τις 2, τότε το bit τίθεται 1, αλλιώς 0. Αν ισχύει ότι $u_1^2 - 4u_0 = 0$, τότε η διακρίνουσα έχει μία λύση και δεν είναι αναγκαία η χρήση αυτού του bit. Στη συνέχεια, αν $v_0 \neq 0$, τότε το δεύτερο bit τίθεται ανάλογα με τη ρίζα που δίνει τη σωστή τιμή για το v_0 στην εξίσωση (2). Αλλιώς, τίθεται η σωστή ρίζα της εξίσωσης (4). Τελικά, η συμπιεσμένη μορφή είναι (u_0, u_1, Bit_1, Bit_2).

Κατά την αποσυμπίεση, είναι γνωστά τα u_1 και u_0, καθώς και όλοι οι συντελεστές f_i από την εξίσωση της καμπύλης και τα Bit_1 και Bit_2. Λύνοντας τη διακρίνουσα του πολυωνύμου $u(x)s(x) + f(x)$ και με τη χρήση του Bit_1, υπολογίζεται το s_0. Με τη χρήση του s_0 και του Bit_2 υπολογίζεται από την εξίσωση (2) το v_0 και στη συνέχεια από την εξίσωση (3) το v_1, αν το $u_0 \neq 0$, αλλιώς υπολογίζεται τοv_1 με τη χρήση του s_0 και του Bit_2 από την (4).

Επομένως, στην περίπτωση των genus 2 καμπυλών που τα πολυώνυμα u και v του divisor έχουν τη μορφή:

$u(x) = x^2 + u_1 * x + u_0$ και $v(x) = v_1 * x + v_0$, αρκεί να αποθηκευτούν και να αποσταλούν οι συντελεστές u_1 και u_0 και ένα επιπλέον byte για τα Bit_1 και Bit_2. Στη συνέχεια τα v_1 και v_0 μπορούν να υπολογιστούν από το πολυώνυμο u. Ο αλγόριθμος της ανακατασκευής παρουσιάζεται αναλυτικά στο βιβλίο. Στο παράρτημα παρατίθεται η υλοποίησή του σε C++.

Τελικά, το αρχικό μέγεθος του divisor από 4 συντελεστές των p bits, μειώνεται σε 2 συντελεστές των p bits. Αυτό είναι ιδιαίτερα χρήσιμο για τη μείωση των μεγεθών των μηνυμάτων που ανταλλάσσονται από τα οχήματα, καθώς τα δημόσια κλειδιά και τα κρυπτογραφημένα κείμενα είναι σε μορφή divisor.

5.3.4 HECC genus 3

5.3.4.1 Βιβλιοθήκη HECC genus 3

Κατά την εκπόνηση της εργασίας δεν βρέθηκε κάποια βιβλιοθήκη που να καλύπτει την κρυπτογραφία HECC με καμπύλες γένους 3. Επομένως, με τη χρήση της libg2hec [43] ως πλαίσιο αναφοράς και τους αλγόριθμους αριθμητικής του βιβλίου των Henri Cohen και Gerhard Frey [44], υλοποιήθηκαν οι αντίστοιχες κλάσεις και μέθοδοι των καμπυλών γένους 3 και των αντίστοιχων divisor. Αρχικά υλοποιήθηκε η κλάση της καμπύλης γένους 3. Πρακτικά, αυτό που άλλαξε από την υλοποίηση της libg2hec είναι ο έλεγχος των βαθμών των πολυωνύμων, ώστε να είναι έγκυρα για καμπύλες γένους 3 και η διαδικασία παραγωγής τυχαίας καμπύλης:

Η μέθοδος g3hcurve::update για τον έλεγχο εγκυρότητας της καμπύλης:

```
//Set is_genus_3
if( deg(fpoly) == 7 && deg(hpoly) <= 3)
        is_genus_3 = TRUE;
else
        is_genus_3 = FALSE;
```

Με τον παραπάνω κώδικα ελέγχεται αν το πολυώνυμο f είναι το πολύ βαθμού 7 (2*g +1) και αν το πολυώνυμο h είναι το πολύ βαθμού 3 (= g). Ο έλεγχος της προϋπόθεσης της καμπύλης να είναι non-singular (βλ. κεφάλαιο 2.2.4) παραμένει ίδια με αυτή της libg2hec με τη χρήση της NTL. Η τυχαία παραγωγή καμπύλης γίνεται με τον ίδιο τρόπο με αυτόν της libg2hec, απλώς αλλάζουν οι βαθμοί από 5 σε 7 του πολυωνύμου f και από 2 σε 4 του πολυωνύμου h της καμπύλης.

Στη συνέχεια υλοποιείται η κλάση των divisor σε καμπύλες γένους 3. Και εδώ υλοποιήθηκε ο έλεγχος του έγκυρου divisor:

```
OK = OK && IsOne( LeadCoeff(upoly) ); // (1)

OK = OK && ( deg(upoly) <= genus ) && ( deg(vpoly) < deg(upoly) );
// (2)

OK = OK && IsZero(( vpoly*(vpoly + curve_g3.get_h())

                            - curve_g3.get_f() ) % upoly ); //(3)
```

Το σημαντικότερο κομμάτι της υλοποίησης, ωστόσο, είναι η αριθμητική των divisor. Η μέθοδοι για τον βαθμωτό πολλαπλασιασμό ενός divisor είναι ανεξάρτητοι του γένους της καμπύλης, επομένως δεν χρήζουν κάποιας ανανέωσης από την libg2hec. Οι μέθοδοι που υποστηρίζονται είναι οι SAM (Square and Multiply), NAF (Non-adjacent Form) και ML (Montgomery's Ladder). Οι αλγόριθμοι που χρήζουν υλοποίησης είναι η «πρόσθεση» και ο «διπλασιασμός» divisor. Η πρόσθεση με βάση τον αλγόριθμο του Cantor (αλγόριθμος 14.7 του HOEHCC [44]), ο οποίος είναι ανεξάρτητος γένους και μπορεί να καλύψει την αριθμητική στις καμπύλες γένους 3. Ωστόσο, είναι αργός και έτσι προτιμήθηκε να χρησιμοποιηθούν οι αλγόριθμοι 14.52 και 14.53 του βιβλίου, οι οποίοι είναι βελτιστοποιημένοι με λιγότερες και πιο αποδοτικές πράξεις. Η υλοποίηση, απλώς, αποτυπώνει τις πράξεις αριθμητικής υπολοίπου των δύο αλγορίθμων σε κώδικα C++.

5.3.4.2 Κρυπτογραφικές τεχνικές

Οι κρυπτογραφικές τεχνικές πρακτικά έχουν την ίδια υλοποίηση, απλώς γίνονται με βάση καμπύλες και divisors γένους 3 με τη χρήση της βιβλιοθήκης που υλοποιήθηκε. Επομένως, η παραγωγή κλειδιών, η κρυπτογράφηση και αποκρυπτογράφηση ElGamal, τα Encodings, οι υπογραφές ElGamal και τα πιστοποιητικά HECQV έχουν ακριβώς την ίδια υλοποίηση με τη χρήση των καμπυλών και divisors γένους 3, ενώ δεν υλοποιήθηκε divisor compression. Η δημιουργία της καμπύλης για όλες τις τεχνικές εκτός των υπογραφών γίνεται με τη χρήση της κλάσεις UnifiedEncoding, όπως και στην HECC genus 2. Η παράμετρος p που επιλέγεται τώρα είναι μεγέθους 86 bits, ώστε το επίπεδο ασφαλείας να είναι 128-bit και με βάση τους περιορισμούς p = 7 mod 8 και p = 3 mod 4:

$$p = 77371252455336267181195223$$

Επιπλέον, η καμπύλη που χρησιμοποιείται επιλέχθηκε από τη δημοσίευση της Annegret Weng [47], η οποία πρότεινε μία κλάση ασφαλών καμπυλών με βάση την μέθοδο CM που προτάθηκε για τις καμπύλες γένους 2. Η καμπύλη που επιλέχθηκε:

$$y^2 = x^7 + x^5 + 6218231719898953 * x^3 + 8683773159487505$$

η οποία παράγει τάξη η οποία είναι «σχεδόν» πρώτη, δηλαδή ισχύει $N = 8q$, όπου το q είναι ένα πρώτος αριθμός 168 bits. Επομένως, ως τάξη (order) εδώ θεωρείται ο αριθμός q.

Οι τελευταίες διαφορές με την υλοποίηση των τεχνικών για γένος 2 είναι η συνάρτηση που αντιστοιχίζει τους divisors σε ακέραιους αριθμούς που χρησιμοποιήθηκε στις υπογραφές ElGamal. Τώρα ως αποτέλεσμα η συνάρτηση επιστρέφει το αποτέλεσμα της πράξης $u_2^2 + u_1^2 + u_0^2 (mod N)$. Επιπλέον, στη διαδικασία text_to_divisor, το μήνυμα χωρίζεται σε 3 ίσα μέρη, αντί για 2, εφόσον πρέπει να παραχθούν 3 σημεία στην καμπύλη. Αφού παραχθούν τα 3 σημεία, η μετατροπή τους σε έναν έγκυρο divisor γίνεται ως εξής:

Οι divisors έχουν τη μορφή: [u(x), v(x)] με $u(x) = x^3 + a * x^2 + b * x + c,\ v(x) = d * x^2 + e * x + f$, επομένως:

```
a = -x1-x2-x3;
b = x1*x2 + x1*x3 + x2*x3;
c = -x1*x2*x3;
ZZ_p e = ((y2-y3)*(x1*x1 - x2*x2) - (y1-y2)*(x2*x2 -
x3*x3))/((x2-x3)*(x1*x1 - x2*x2) - (x1-x2)*(x2*x2 - x3*x3));
ZZ_p d = (y1 - y2 - e*(x1-x2))/(x1*x1 - x2*x2);
ZZ_p f = y3 - e*x3 - d*x3*x3;
```

Η αντίστροφη διαδικασία υλοποιείται με απλό τρόπο:

```
vec_ZZ_p roots = FindRoots(u);
x1 = roots[0];
x2 = roots[1];
x3 = roots[2];
y1 = eval(v, x1);
y2 = eval(v, x2);
y3 = eval(v, x3);
```

5.4 Υλοποίηση οδικού δικτύου

Για την υλοποίηση του οδικού δικτύου χρησιμοποιήθηκε το εργαλείο SUMO. Αρχικά έγινε εξαγωγή μίας πραγματικής περιοχής από τον χάρτη σε ένα αρχείο osm. Αυτό, με τη χρήση του εργαλείου netconvert του SUMO μετατράπηκε σε ένα xml αρχείο για τη χρήση του στην παραγωγή του δικτύου και των διαδρομών. Στη συνέχεια, με τη χρήση του εργαλείου randomTrips του SUMO παρήχθησαν τυχαίες διαδρομές κόμβων πάνω στον χάρτη και αποθηκεύτηκαν σε ένα xml αρχείο. Με τη βοήθεια των παραμέτρων του randomTrips, επιλέχθηκε η προσομοίωση να τελειώνει γύρω στα 400 δευτερόλεπτα και να παράγεται ένα όχημα ανά 0.5 δευτερόλεπτα. Στη συνέχεια, εκτελέστηκε η προσομοίωση κίνησης με βάση τον χάρτη και τις διαδρομές που παρήχθησαν και έγινε εξαγωγή ενός αρχείου sumoTrace.xml με την ολοκληρωμένη προσομοίωση. Τέλος, αυτό το αρχείο μετατράπηκε σε ένα αρχείο tcl το οποίο είναι κατάλληλο για χρήση με το NS-2 και με το NS-3 με τη χρήση του εργαλείου traceExporter.

Το αρχείο φορτώθηκε στην προσομοίωση του NS-3 με τη χρήση της κλάσης Ns2MobilityHelper, η οποία εγκαθιστά τους κόμβους και την κίνηση τους. Τα οχήματα που εισέρχονται στην προσομοίωση είναι συνολικά 63, ενώ προστέθηκε ένας επιπλέον κόμβος χειροκίνητα στην προσομοίωση για να κατέχει τον ρόλο της RSU. Τοποθετήθηκε περίπου στο μέσο του χάρτη. Ωστόσο, αντιμετωπίστηκε ένα πρόβλημα κατά την τοποθέτηση των οχημάτων, καθώς η κλάση Ns2MobilityHelper εγκαθιστά όλα τα οχήματα από την αρχή της προσομοίωσης και έτσι λαμβάνουν πακέτα, χωρίς στην πραγματικότητα να έχουν εισέλθει ακόμα στην προσομοίωση. Κάθε όχημα εισέρχεται και εξέρχεται από την προσομοίωση μία συγκεκριμένη χρονική στιγμή. Οι χρονικές στιγμές ακολουθούν κατανομή Poisson. Επομένως, με

τη χρήση της κλάσης Ns2NodeUtility [48] είναι εφικτό να ληφθούν από το αρχείο tcl οι ακριβείς χρόνοι των εισερχόμενων και εξερχόμενων οχημάτων, με αποτέλεσμα η επικοινωνία να ξεκινήσει και να σταματήσει με βάση αυτούς τους χρόνους και να μην αρχίσουν να επικοινωνούν όλα μαζί πριν πραγματικά εισέλθουν στην περιοχή.

Σε κάθε κόμβο εγκαταστάθηκαν συσκευές WAVE με τη χρήση της κλάσης YansWavePhyHelper που παρέχεται από της βοηθητικές συναρτήσεις του NS-3.

```
NetDeviceContainer devices = waveHelper.Install (wavePhy,
waveMac, nodes);
```

Το φυσικό στρώμα wavePhy ρυθμίστηκε με βάση το YansWavePhyHelper και το waveMac με την κλάση QosWaveMacHelper. Επιπλέον, τέθηκε η ελάχιστη και η μέγιστη ισχύ εκπομπής στα 8 dBm και 33 dBm αντίστοιχα με βάση τις διεθνείς προδιαγραφές. Σε κάθε εκπομπή πακέτου επιλέγεται η μέγιστη ισχύ και η μέγιστη προτεραιότητα.

5.5 Υλοποίηση δικτύου επικοινωνίας

Η επικοινωνία των οχημάτων και της RSU με σκοπό την υλοποίηση του ασφαλούς σχήματος 9 με τις επιθυμητές κρυπτογραφικές τεχνικές έγινε με τη εξ' ολοκλήρου με τη χρήση του NS-3 σε C++ με τη βοήθεια των βιβλιοθηκών Crypto++, libg2hec, NTL και g3hec που υλοποιήθηκε κατά τη διάρκεια της εργασίας και αναλύθηκε στο κεφάλαιο 5.3.4. Στο πλαίσιο της εργασίας, δεν υλοποιήθηκε η ανταλλαγή και χρήση ψευδωνύμων, καθώς και η επικοινωνία την CA για ανανεώσεις και εκδόσεις πιστοποιητικών.

Η επικοινωνία υλοποιήθηκε με τη χρήση πακέτων στο στρώμα WAVE για την εξαγωγή των μετρικών χωρίς την πρόσθετη χρονική επεξεργασία των ανώτερων στρωμάτων δικτύου και εφαρμογής. Επιπλέον, εγκαταστάθηκαν συναρτήσεις callbacks, οι οποίες καλούνται όταν λαμβάνεται ένα πακέτο σε κάποιον κόμβο (Rx Callbacks) και ανάλογα με το στάδιο της επικοινωνίας (Join, Accept, Extract_Symmetric, Receive GL Proof, Receive Vehicle Information) επεξεργάζονται το εισερχόμενο πακέτο και συνεχίζουν σε περαιτέρω ενέργειες, όπως το να εκπέμψουν ένα πακέτο απάντησης αν χρειάζεται. Οι απαντήσεις δρομολογούνται, ώστε να εκτελεστούν τυχαία σε ένα διάστημα 0-3 δευτερολέπτων με σκοπό την καλύτερη επίβλεψη της επικοινωνίας. Οι διαφορετικές καταστάσεις ορίστηκαν σε μία κλάση enumeration και συγκεκριμένα είναι οι εξής:

```
enum ProtocolVEH {
    RECEIVE_CERT,
    RECEIVE_ACCEPT_KEY,
    ON_SYMMETRIC_ENC,
    GROUP_LEADER_INFORM,
    IS_GROUP_LEADER,
    RECEIVE_ACCEPT_GL,
    ON_SYMM_GL,
```

```
    INFORM_MSG
};
```

Οι καταστάσεις αφορούν τα οχήματα, την RSU και τον Group Leader. Επιπλέον, ο κάθε τύπος κόμβου (Vehicle, RSU, GL) έχει μία δομή, η οποία αποθηκεύει τις απαραίτητες πληροφορίες για την επικοινωνία.

```
struct RSU_data
struct Vehicle_data
struct GroupLeader_data
```

Η RSU αποθηκεύει την κατάσταση όλων των οχημάτων της περιοχής, το ζεύγος κλειδιών της, της CA και το δημόσιο κλειδί των οχημάτων που έχουν εισέλθει στην περιοχή και το πιστοποιητικό της. Αποθηκεύει τον συνολικό αριθμό των οχημάτων που έχουν κάνει Join, το αναγνωριστικό του GL, τα συμμετρικά κλειδιά που έχουν ανταλλαχθεί με τα οχήματα και τις παραμέτρους των καμπυλών που χρησιμοποιούνται για κρυπτογραφία. Το όχημα αποθηκεύει τα κλειδιά του και τα κλειδιά της CA, το δημόσιο κλειδί της RSU και του GL αν υπάρχει, το πιστοποιητικό του, την κατάστασή του, το συμμετρικό κλειδί και τον πίνακα IV και τις παραμέτρους των καμπυλών. Τα δεδομένα του GL περιλαμβάνουν τα Vehicle_data του και τα αντίστοιχα δεδομένα που αποθηκεύονται στην RSU για κάθε όχημα.

Η επικοινωνία ξεκινάει με την RSU να εκπέμπει το πιστοποιητικό της κάθε 2 δευτερόλεπτα. Όλα τα οχήματα βρίσκονται σε κατάσταση RECEIVE_CERT. Μόλις λάβουν το πιστοποιητικό το ελέγχουν, λαμβάνουν το κλειδί της RSU, το αποθηκεύουν και δημιουργούν το πακέτο Join το οποίο και στέλνουν πίσω στην RSU και μεταβαίνουν σε κατάσταση RECEIVE_ACCEPT_KEY. Η RSU επεξεργάζεται την απάντηση και αν είναι έγκυρη αλλάζει την κατάσταση του οχήματος αρχικά σε RECEIVE_ACCEPT_KEY και μετά σε ON_SYMMETRIC_ENC, αφού στείλει πίσω την απάντηση Accept στο όχημα. Το όχημα λαμβάνει το συμμετρικό κλειδί από το πακέτο Accept και μεταβαίνει σε κατάσταση ON_SYMMETRIC_ENC. Στη συγκεκριμένη προσομοίωση, η πυκνότητα των οχημάτων σε μία περιοχή μεγαλώνει μετά τα 120 δευτερόλεπτα, όπου δημιουργείται μποτιλιάρισμα. Επομένως, αποφασίζεται από την RSU να εκλεχθεί ένας GL με τυχαίο τρόπο, ο οποίος θα αναλάβει να εξυπηρετεί τα οχήματα με ζυγό ID από εκείνη τη χρονική στιγμή και πέρα. Του στέλνει το μήνυμα GL Proof of Leadership και μεταβαίνει την κατάστασή του αρχικά σε GROUP_LEADER_INFORM και στη συνέχεια σε IS_GROUP_LEADER. Εδώ, να σημειωθεί πως επειδή δεν ήταν εφικτό να κρυπτογραφηθεί το μήνυμα Proof of Leadership με το μυστικό κλειδί, καθώς αυτή η δυνατότητα δεν παρέχεται από τις κρυπτογραφικές τεχνικές ECC, HECC, αποφασίστηκε να χρησιμοποιηθεί το πιστοποιητικό του GL, υπογεγραμμένο με το δημόσιο κλειδί της RSU. Αφού ο GL λάβει το μήνυμα και το αποκρυπτογραφήσει το αναμεταδίδει κάθε 2 δευτερόλεπτα στο δίκτυο και μεταβαίνει και αυτό σε κατάσταση IS_GROUP_LEADER. Τα ήδη υπάρχοντα οχήματα, αλλά και αυτά που εισέρχονται αργότερα με ζυγό ID ακολουθούν τη διαδικασία Join και Accept με τον GL (χρησιμοποιούνται οι καταστάσεις RECEIVE_ACCEPT_GL και ON_SYMM_GL). Τέλος, τα οχήματα μεταδίδουν σε τυχαία χρονική στιγμή ένα μήνυμα

Inform, όταν έχουν λάβει συμμετρικό κλειδί. Ο GL συγκεντρώνει όλα τα μηνύματα Inform και τα στέλνει ομαδοποιημένα στην RSU.

Τα μηνύματα που στέλνονται είναι διαφορετικά ανάλογα με την κατάσταση που βρίσκονται. Το πιστοποιητικό της RSU και το Proof of Leadership έχουν την εξής μορφή:

SENDER ID: 1 Byte	RECEIVE_CERT TYPE: 1 Byte	CERTIFICATE: Depends on ECC, HECC	CURVE PARAMETERS (OPTIONAL): Depends on ECC, HECC	SIGNATURE (FOR GL): Depends on ECC, HECC

Εικόνα 20: Δομή μηνύματος πιστοποιητικού RSU και GL Proof of Leadership

Τα υπόλοιπα μηνύματα:

SENDER ID: 1 Byte	TYPE: 1 Byte	Encrypted Message: Depends on ECC, HECC, AES and message type	Certificate (OPTIONAL): Depends on ECC, HECC	Signature: Depends on ECC, HECC

Εικόνα 21: Δομή μηνυμάτων ασφαλείας

Τα μηνύματα δημιουργούνται και στέλνονται ως byte buffers (uint8_t *). Να σημειωθεί πως, λόγω της χρήση διαφορετικής καμπύλης στην περίπτωση των υπογραφών πρέπει να χρησιμοποιηθούν διαφορετικά κλειδιά για τις υπογραφές. Για την μη επιβάρυνση του δικτύου, λόγω αυτής της ιδιαιτερότητας επιλέγεται το κλειδί των υπογραφών να είναι σταθερό για όλους, υποθέτοντας πως η υπογραφή γίνεται με το ίδιο κλειδί των υπόλοιπων κρυπτογραφικών τεχνικών.

5.6 Υλοποίηση μοντέλου ενέργειας

Το NS-3 παρέχει μηχανισμούς για την προσομοίωση της κατανάλωσης ενέργειας των δικτυακών συσκευών. Συγκεκριμένα, χρησιμοποιήθηκε το μοντέλο WiFi Radio Energy Model του NS-3, το οποίο υπολογίζει μείωση της μπαταρίας της συσκευής ανάλογα με το ρεύμα που απαιτεί κάθε μία από τις καταστάσεις του WiFi (Idle, CcaBusy, Tx, Rx, ChannelSwitch, Sleep, off). Το μοντέλο αναλαμβάνει να υπολογίζει σε κάθε αλλαγή της κατάστασης την ενέργεια που καταναλώθηκε στο χρονικό διάστημα που πέρασε.

Το μοντέλο WiFi Radio Energy Model χρειαζόταν μία μικρή τροποποίηση στην κλάση WifiRadioEnergyModelHelper, ώστε να λειτουργήσει σωστά με συσκευές τύπου WAVE. Συγκεκριμένα, για να εξαχθεί το φυσικό στρώμα WiFi από την συσκευή WAVE και να εγκατασταθεί το μοντέλο ενέργειας σε αυτή προστέθηκαν οι παρακάτω γραμμές κώδικα:

```
Ptr<WaveNetDevice> waveDevice = DynamicCast<WaveNetDevice>
(device);
Ptr<WifiPhy> wifiPhy = waveDevice->GetPhy (0);
wifiPhy->SetWifiRadioEnergyModel (model);
```

Στη συνέχεια, επιλέχθηκε οι κόμβοι να έχουν αυθαίρετα μεγάλη ποσότητα αρχικής ενέργειας των 1000 Joule στην μπαταρία, αφού η παρούσα εργασία μελετά την κατανάλωση ενέργειας, άρα τη διαφορά της ενέργειας την μπαταρία πριν και μετά από κάθε επικοινωνία.

```
BasicEnergySourceHelper energyHelper;
energyHelper.Set("BasicEnergySourceInitialEnergyJ", DoubleValue
(1000.0));
*Vehicle_sources = energyHelper.Install(nodes);
WaveRadioEnergyModelHelper waveEnergyHelper;

DeviceEnergyModelContainer deviceModels =
waveEnergyHelper.Install(devices, *Vehicle_sources);
```

Η εναπομένουσα ενέργεια στην μπαταρία κάθε οχήματος αποθηκεύεται σε έναν πίνακα prev_energy και σε κάθε στάδιο επικοινωνίας υπολογίζεται η κατανάλωση ως εξής:

```
consumption = prev_energy[vid] - Vehicle_sources->Get(vid)-
>GetRemainingEnergy()
prev_energy[vid] = Vehicle_sources->Get(vid)-
>GetRemainingEnergy();
```

Κεφάλαιο 6: Πειραματική Αξιολόγηση και Συμπεράσματα

Στο κεφάλαιο αυτό παρουσιάζονται και σχολιάζονται τα πειραματικά αποτελέσματα της εργασίας με τη χρήση των τριών αλγόριθμων ασύμμετρης κρυπτογράφησης.

Τα οχήματα που χρησιμοποιήθηκαν για την προσομοίωση της επικοινωνίας είναι 63 σε αριθμό, ενώ χρησιμοποιήθηκε 1 RSU, η οποία καλύπτει την περιοχή. Επιπλέον, τα μηνύματα μεταδίδονταν με τη μέγιστη δυνατή ενέργεια και προτεραιότητα και για τη μείωση των συγκρούσεων από ταυτόχρονη μετάδοση επιλέχθηκε να υλοποιηθεί μία απλή υποχώρηση των απαντήσεων των οχημάτων έως και 3 δευτερόλεπτα. Το διάστημα είναι αρκετά μεγάλο για πραγματικά συστήματα, ωστόσο χρησιμοποιήθηκε για την καλύτερη επίβλεψη της επικοινωνίας. Εξ' άλλου οι μετρήσεις γίνονται στις κρυπτογραφικές τεχνικές. Η συγκεκριμένη προσέγγιση αυξάνει το συνολικό χρόνο και επομένως αυξάνει και την ενέργεια που καταναλώνεται στο διάστημα, μιας και αυτή εξαρτάται από τον χρόνο λειτουργίας. Έτσι, σε κάποια μέρη της επικοινωνίας παρατηρείται μεγαλύτερη κατανάλωση από άλλα.

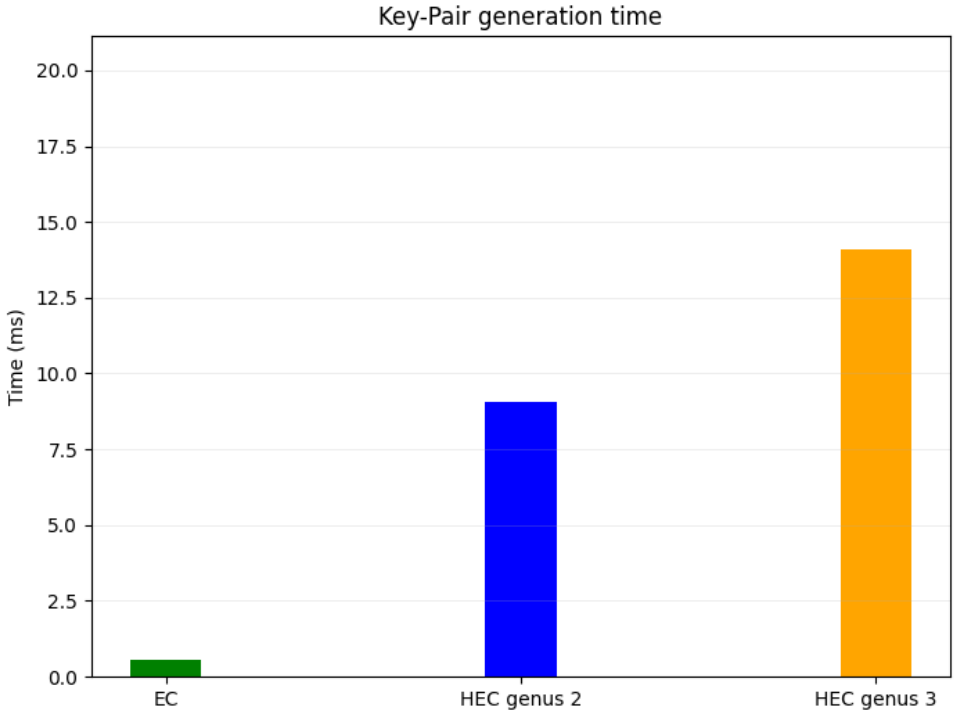
Για τη μέτρηση του χρόνου χρησιμοποιήθηκε η συνάρτηση chrono::high_resolution_clock::now() στην αρχή και στο τέλος κάθε υπολογισμού. Στα διαγράμματα παρατίθεται ο μέσος όρος των χρόνων και των ενεργειών. Η μέτρηση των μεγεθών έγινε με τη μέτρηση του byte buffer πριν σταλεί από τον κόμβο. Επομένως, στις μετρήσεις δεν συμπεριλαμβάνεται το μέγεθος της επικεφαλίδας που προσθέτει το WAVE.

Οι προσομοιώσεις πραγματοποιήθηκαν σε εικονικό περιβάλλον Linux Ubuntu 20.04.6 με host Windows 10, διαθέσιμη μνήμη 8 GB και 4 επεξεργαστές τύπου Intel Core i5-10300H με συχνότητα 2.50 GHz. Επίσης χρησιμοποιήθηκαν:

- NS-3.30
- SUMO 1.16.0
- NTL 5.5

6.1 Πειραματική αξιολόγηση

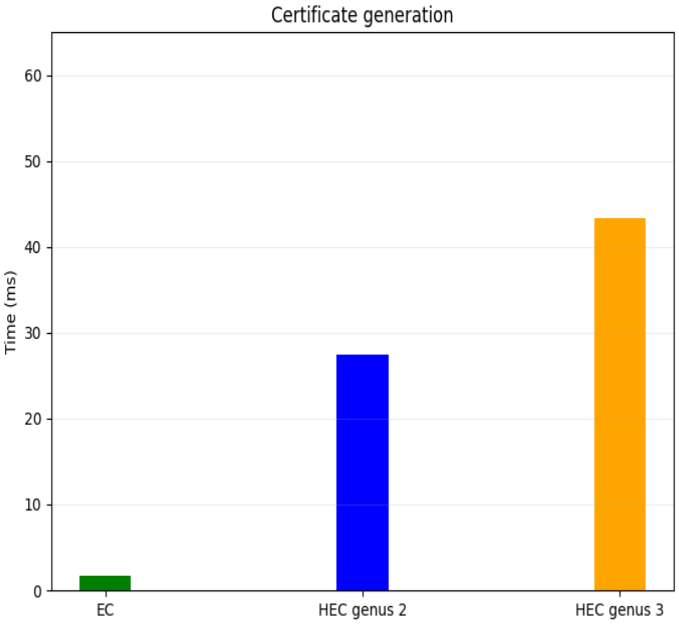
6.1.1 Χρόνοι παραγωγής κλειδιών



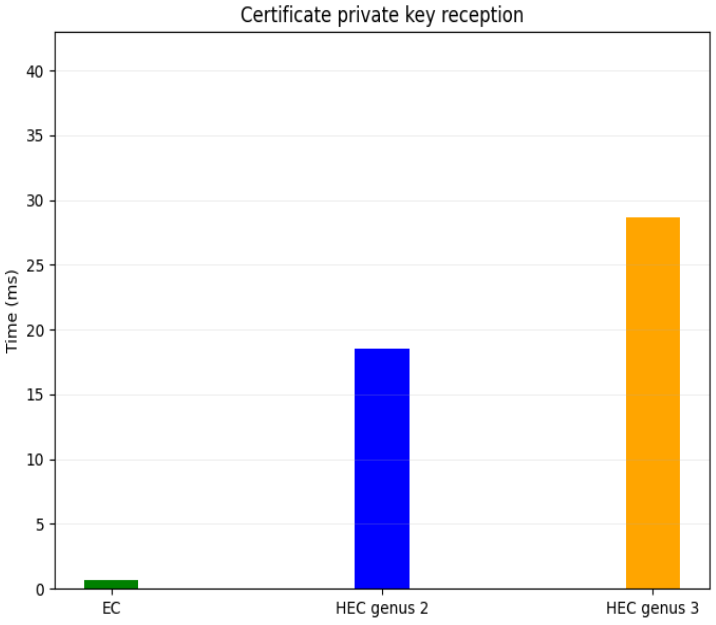
Εικόνα 22: Χρόνοι παραγωγής κλειδιών (ms)

Παρατηρείται από το παραπάνω διάγραμμα πως στο ίδιο επίπεδο ασφαλείας των 128-bit η παραγωγή κλειδιών με βάση την ECC είναι σχεδόν 16 φορές πιο γρήγορη από την HECC genus 2 και 27 φορές πιο γρήγορη από την HECC genus 3.

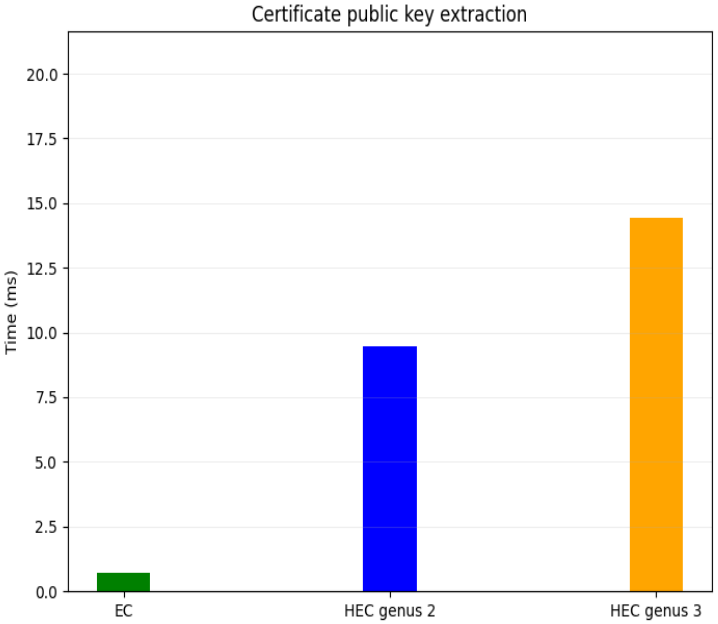
6.1.2 Χρόνοι παραγωγής πιστοποιητικών και εξαγωγής κλειδιών



Εικόνα 24: Χρόνος παραγωγής πιστοποιητικού (ms)



Εικόνα 23: Χρόνος εξαγωγής ιδιωτικού κλειδιού (ms)

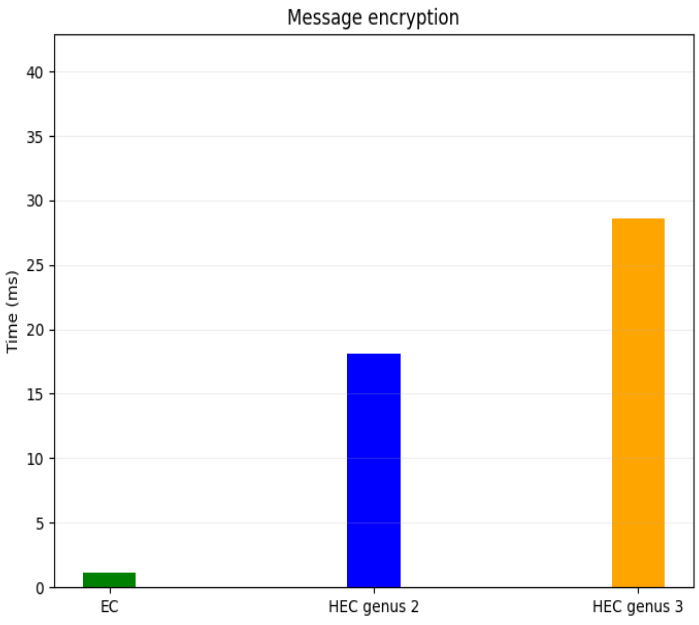


Εικόνα 25: Χρόνος εξαγωγής δημόσιου κλειδιού (ms)

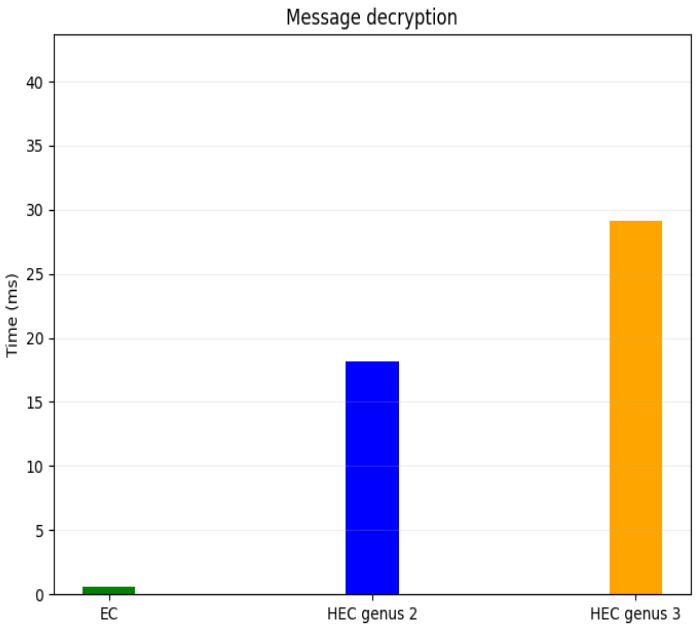
Παρατηρείται πως τα αποτελέσματα είναι σχεδόν ίδια και στη διαδικασία παραγωγής πιστοποιητικών και εξαγωγής κλειδιών με βάση το σχήμα ECQV από αυτά. Η παραγωγή και εξαγωγή με βάση την ECC είναι σχεδόν 15 φορές πιο γρήγορες από την HECC genus 2 και 25 φορές πιο γρήγορη από αυτές της HECC genus 3. Παρατηρείται πως η παραγωγή είναι η πιο αργή διαδικασία, ακολουθεί η

διαδικασία εξαγωγής του μυστικού κλειδιού και τέλος πιο γρήγορη διαδικασία είναι αυτή της εξαγωγής του δημοσίου κλειδιού. Συγκεκριμένα είναι σχεδόν 3 φορές πιο γρήγορη από αυτή της παραγωγής.

6.1.3 Χρόνοι κρυπτογράφησης και αποκρυπτογράφησης



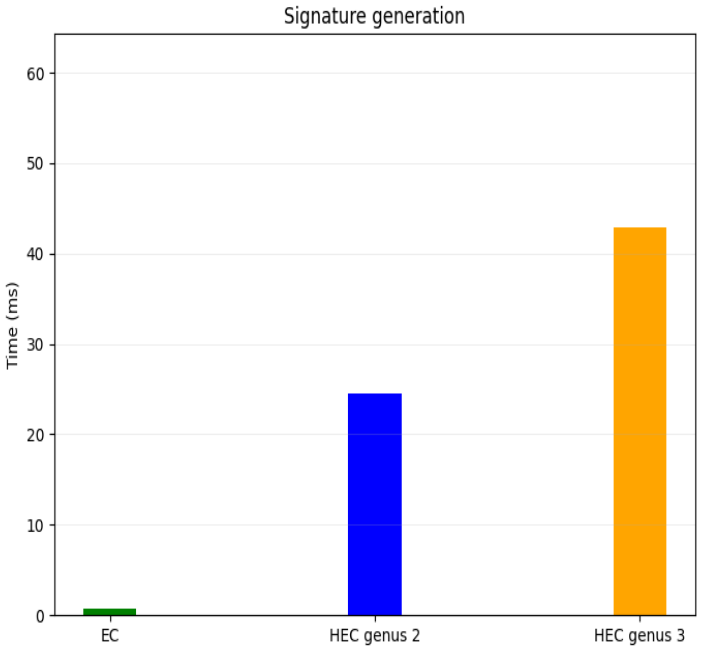
Εικόνα 27: Χρόνος κρυπτογράφησης μηνύματος (ms)



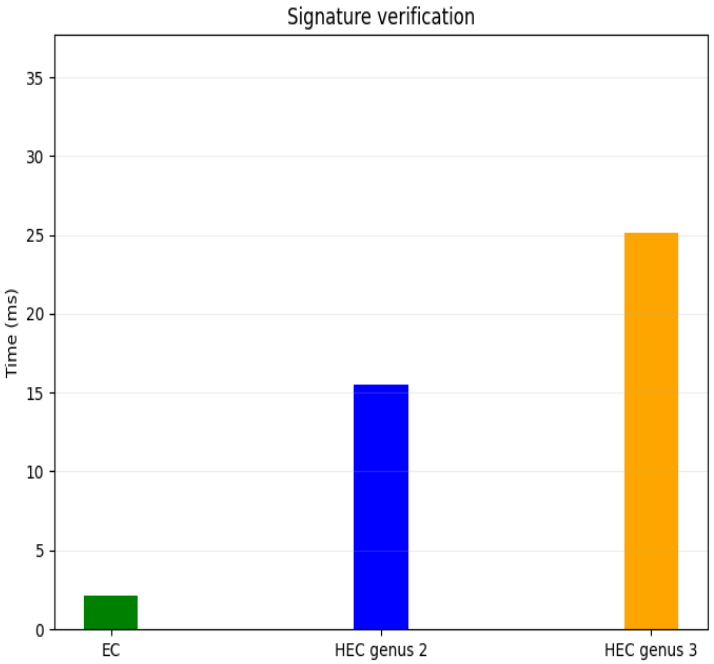
Εικόνα 26: Χρόνος αποκρυπτογράφησης μηνύματος (ms)

Ξανά, η κρυπτογράφηση και η αποκρυπτογράφηση με βάση την ECC είναι αρκετά πιο γρήγορη από τις άλλες δύο κρυπτογραφικές τεχνικές (13 φορές από την HECC genus 2 και 21 φορές από την HECC genus 3). Παρατηρείται, πως η αποκρυπτογράφηση είναι ελαφρώς γρηγορότερη διαδικασία από την κρυπτογράφηση.

6.1.4 Χρόνοι παραγωγής και επικύρωσης υπογραφής



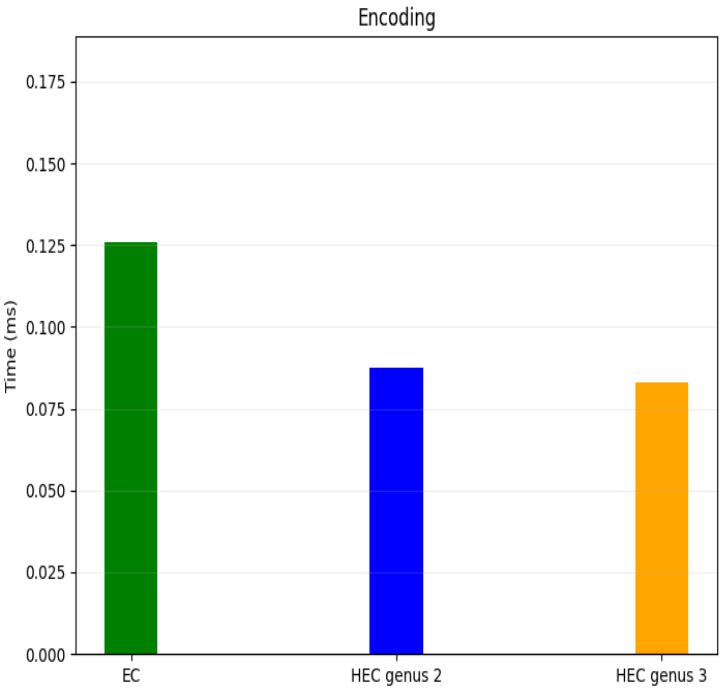
Εικόνα 29: Χρόνος παραγωγής υπογραφής (ms)



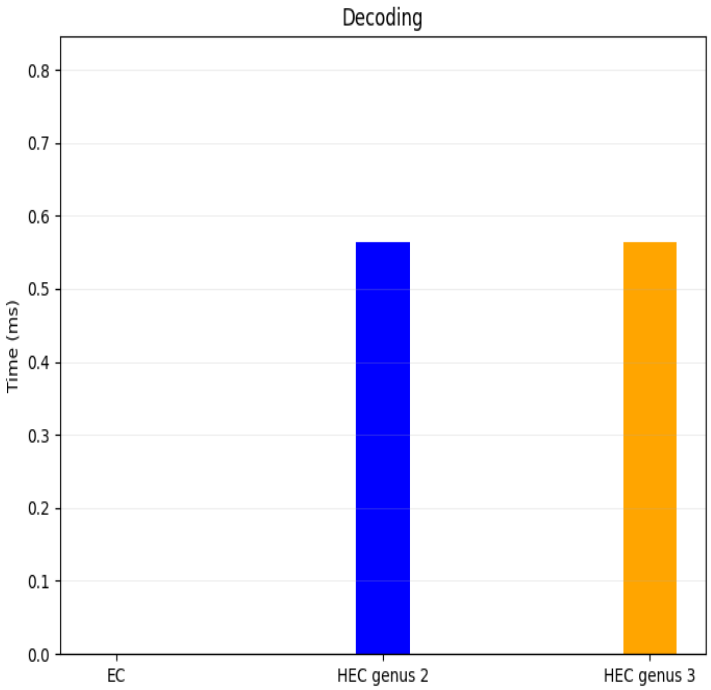
Εικόνα 28: Χρόνος επικύρωσης υπογραφής (ms)

Ξανά η ECC καταφέρνει να επισκιάσει σε χρόνο τις άλλες δύο υλοποιήσεις. Ωστόσο, η διαφορά κατά την επικύρωση της υπογραφής είναι μικρότερη συγκριτικά με άλλες κρυπτογραφικές τεχνικές. Βέβαια, αυτό οφείλεται και στο γεγονός της αναγκαστικής επιλογής διαφορετικών καμπυλών για τις υπογραφές με βάση τις τεχνικές HECC, οι οποίες είναι στο επίπεδο ασφαλείας 84 bit, ενώ η καμπύλη στην ECC προσδίδει ασφάλεια επιπέδου 128-bit.

6.1.5 Χρόνοι κωδικοποίησης και αποκωδικοποίησης μηνύματος



Εικόνα 31: Χρόνος κωδικοποίησης μηνύματος (ms)



Εικόνα 30: Χρόνος αποκωδικοποίησης μηνύματος (ms)

Εδώ παρατηρείται μία αλλαγή του μοτίβου που διακρινόταν προηγουμένως. Η κωδικοποίηση με βάση τον αλγόριθμο Koblitz είναι ελαφρώς πιο αργή από την κωδικοποίηση με βάση τους αλγορίθμους που αναπτύχθηκαν για HECC genus 2 και 3, ενώ η κωδικοποίηση με HECC genus 3 είναι πιο γρήγορη από αυτή της HECC genus 2. Από την άλλη, η αποκωδικοποίηση με χρήση του αλγόριθμου Koblitz έχει σχεδόν μηδενικό χρόνο, ενώ οι άλλοι δύο αλγόριθμοι έχουν σχεδόν ίδιους χρόνους αποκωδικοποίησης.

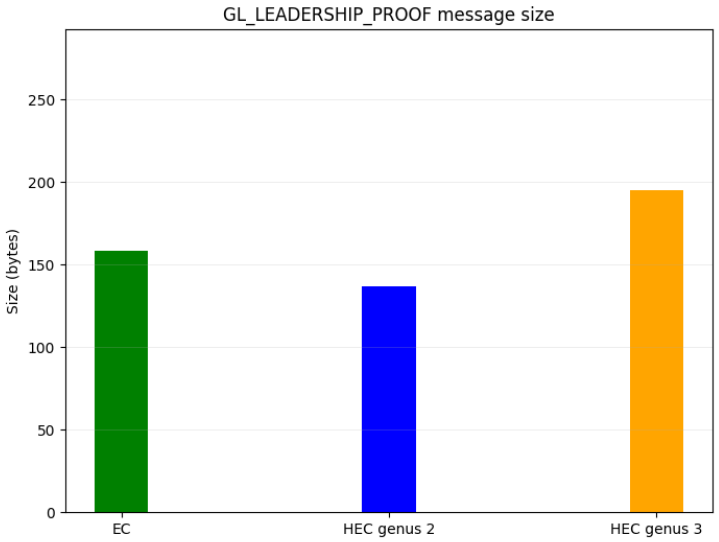
6.1.6 Μέγεθος μηνυμάτων

Για την προσομοίωση της επικοινωνίας χρησιμοποιήθηκαν κάποιοι συγκεκριμένοι τύποι μηνυμάτων με βάση τον αλγόριθμο 9. Παρακάτω αναλύονται:

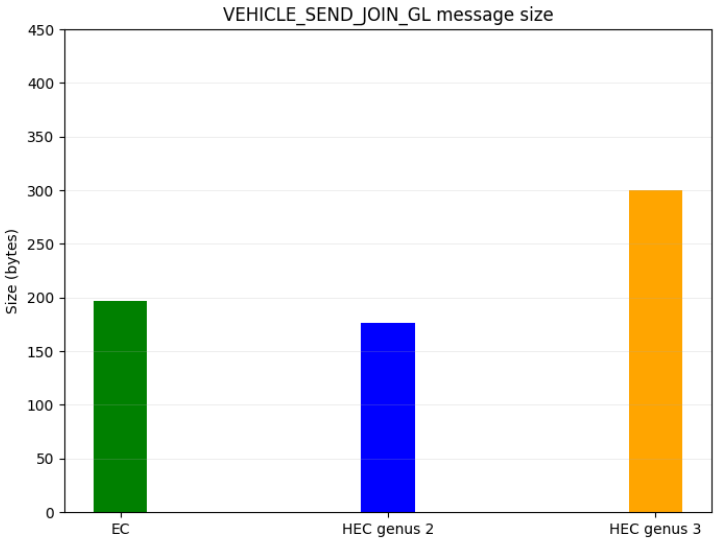
- **RSU_CERT_BROADCAST**: Το πιστοποιητικό που η RSU εκπέμπει περιοδικά.
- **VEHICLE_SEND_JOIN_RSU**: Το μήνυμα Join που στέλνει ένα όχημα στην RSU, όταν επιθυμεί να εγγραφεί στην περιοχή της.
- **RSU_ACCEPT**: Η απάντηση της RSU στο μήνυμα Join του οχήματος που εμπεριέχει και το συμμετρικό κλειδί κρυπτογράφησης.
- **RSU_INFORM_LEADER**: Το μήνυμα που στέλνει η RSU, ώστε να ενημερώσει ένα όχημα ότι επιλέχθηκε ως Group Leader. Φέρει την απόδειξη αρχηγίας (Proof of Leadership).

- **GL_LEADERSHIP_PROOF**: Το μήνυμα που εκπέμπει περιοδικά ο GL, για να αποδείξει ότι είναι έγκυρος και να προσκαλέσει τα οχήματα να κάνουν Join σε αυτόν.
- **VEHICLE_SEND_JOIN_GL**: Αντίστοιχο με το μήνυμα VEHICLE_SEND_JOIN_RSU
- **GL_ACCEPT**: Αντίστοιχο του RSU_ACCEPT.
- **VEHICLE_INFORM**: Το μήνυμα που στέλνει ένα όχημα για να ενημερώσει για την τοποθεσία του. Επιλέχθηκε τυπικά ως ένα μήνυμα ενημέρωσης για την αξιολόγηση του συστήματος.

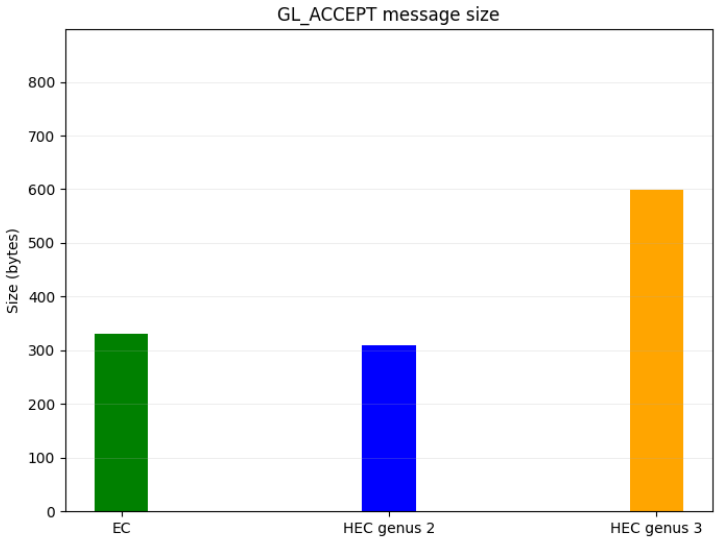
Παρακάτω παρατίθενται τα μεγέθη των μηνυμάτων που δημιουργούνται με βάση τους 3 διαφορετικούς αλγόριθμους ασύμμετρης κρυπτογράφησης, όπως αυτά μετρήθηκαν στην προσομοίωση:



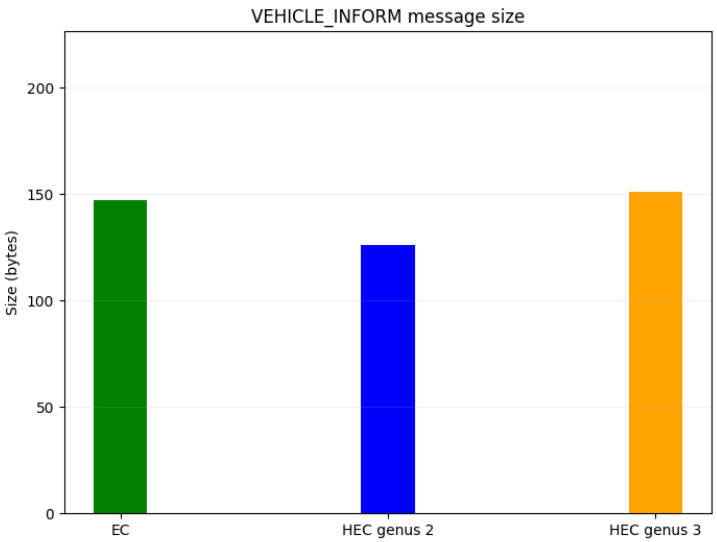
Εικόνα 39: Μέγεθος GL_LEADERSHIP_PROOF σε (bytes)



Εικόνα 38: Μέγεθος VEHICLE_SEND_JOIN_GL σε (bytes)



Εικόνα 37: Μέγεθος GL_ACCEPT σε (bytes)



Εικόνα 36: Μέγεθος VEHICLE_INFORM σε (bytes)

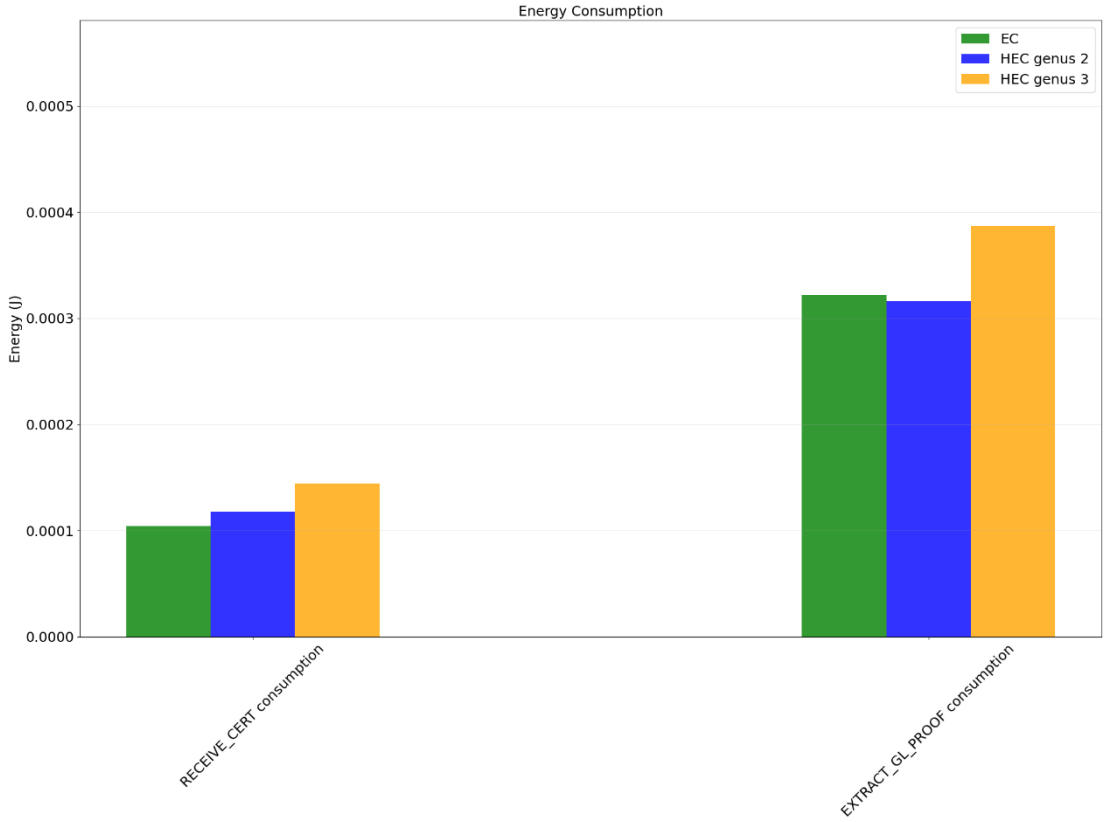
Σε γενικές γραμμές παρατηρείται πως τα μηνύματα με χρήση ECC και HECC genus 2 είναι σχεδόν ίδια, με μικρή διαφορά υπέρ του HECC genus 2, η οποία προκύπτει από το μέγεθος της υπογραφής που έχει υλοποιηθεί για μικρότερο επίπεδο ασφαλείας. Αυτό ήταν αναμενόμενο στο ίδιο επίπεδο ασφαλείας με τη χρήση της συμπίεσης. Από την άλλη, καθώς δεν έχει υλοποιηθεί συμπίεση στη HECC genus 3 τα μηνύματα είναι μεγαλύτερα. Επιπλέον, το μήνυμα RSU_CERT_BROADCAST στην περίπτωση των HECC περιέχει και επιπλέον πληροφορία για τις παραμέτρους της καμπύλης, ενώ στην ECC επιλέγεται απλώς μία καμπύλη και οι κλάσεις της Crypto++ διαχειρίζονται όλες τις παραμέτρους στο υπόβαθρο.

6.1.7 Κατανάλωση ενέργειας

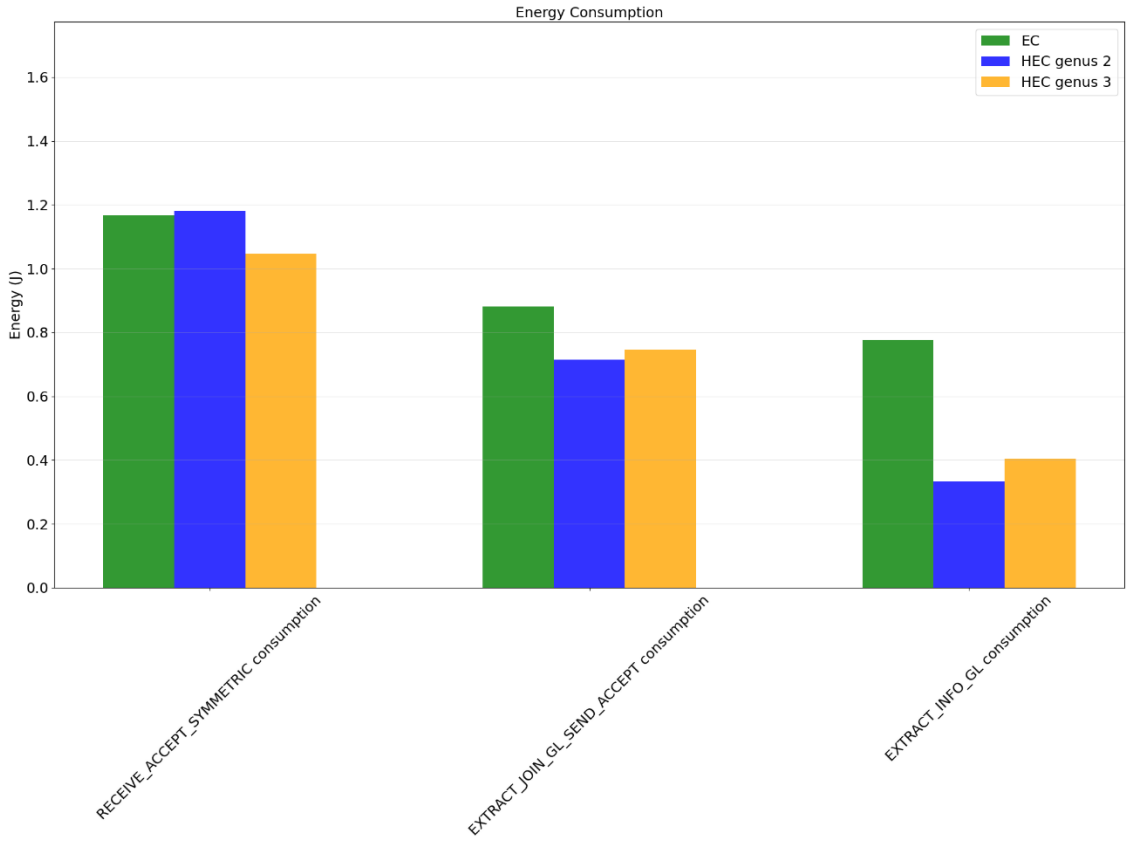
Η κατανάλωση ενέργειας επιλέχθηκε να μετρηθεί σε στάδια, τα οποία ομαδοποιούνται σε παρόμοιους χρόνους, ώστε να υπάρχει μία καλύτερη σύγκριση. Η κατανάλωση ενέργειες έχει σημασία να μετρηθεί στα οχήματα και όχι στην υποδομή. Τα στάδια είναι τα εξής:

- **RECEIVE_CERT**: Περιγράφει απλώς τη διαδικασία λήψης και επεξεργασίας του πιστοποιητικού της RSU (διαδικασία πολύ μικρού χρόνου).
- **EXTRACT_GL_PROOF**: Περιγράφει τη διαδικασία λήψης και επεξεργασίας του αποδεικτικού αρχηγίας του GL (διαδικασία πολύ μικρού χρόνου).
- **RECEIVE_ACCEPT_SYMMETRIC**: Περιγράφει τη διαδικασία αποστολής του μηνύματος Join και λήψης και επεξεργασίας της απάντησης Accept από την RSU.
- **EXTRACT_JOIN_SEND_ACCEPT**: Περιγράφει τη διαδικασία επεξεργασίας του μηνύματος Join που έλαβε ο GL από ένα όχημα και της αποστολής της απάντησης Accept.
- **EXTRACT_INFO_GL**: Περιγράφει τη διαδικασία εξαγωγής ενός μηνύματος Inform που λαμβάνει ο GL από ένα όχημα.

Στα διαγράμματα αποτυπώνεται ο μέσος όρος κατανάλωσης κάθε σταδίου σε Joule:



Εικόνα 40: Κατανάλωση ενέργειας σε διαδικασίες πολύ μικρού χρόνου



Εικόνα 41: Κατανάλωση ενέργειας διαδικασιών

Παρατηρείται πως τη μεγαλύτερη ενέργεια καταναλώνει η HECC genus 3 για τις διαδικασίες πολύ μικρού χρόνου, ενώ στις υπόλοιπες η ECC καταναλώνει τη μεγαλύτερη από τις 3. Γενικότερα, είναι γνωστό πως η κατανάλωση ενέργειας σχετίζεται με τα μεγέθη των μηνυμάτων, καθώς μεγάλα μηνύματα που απαιτούν κατάτμηση δημιουργούν περισσότερα state switches στη συσκευή εκπομπής. Ωστόσο, λόγων των αλγορίθμων που επιλέχθηκαν τα μηνύματα είναι μικρά και στο επίπεδο του WAVE δεν απαιτούν κατάτμηση.

6.2 Συμπεράσματα

Με βάση τα πειραματικά αποτελέσματα που παρουσιάστηκαν στο κεφάλαιο 6.1 εξάγονται κάποια συμπεράσματα για τη χρήση των τεχνικών κρυπτογράφησης ECC και HECC στο σχήμα ασφαλούς αυθεντικοποίησης και μετάδοσης μηνυμάτων σε VANETs των Mistaheeri κ.α. [28]. Αρχικά, το επίπεδο ασφαλείας που μελετήθηκε η εφαρμογή των αλγορίθμων είναι το επίπεδο 128-bit που είναι επιθυμητό για σχήματα ασφαλείας με περιορισμένες δυνατότητες και επίπεδο 84-bit στις υπογραφές των HECC για ερευνητικό σκοπό.

Τις καλύτερες επιδόσεις τις πετυχαίνει η ECC με διαφορά στους χρόνους. Ήδη οι χρόνοι υπολογισμών είναι στην ECC είναι bottleneck συγκριτικά με άλλους αλγόριθμους κρυπτογράφησης, όπως είναι ο RSA και επομένως, η χρήση των HECC με τις παρούσες υλοποιήσεις δεν καλύπτει τις προϋποθέσεις ταχύτητας. Συγκεκριμένα, συγκριτικά με την υλοποίηση του σχήματος [28] με RSA, η ECC καθυστερεί περισσότερο στις υπογραφές, ωστόσο είναι πιο γρήγορη στην κρυπτογράφηση και αποκρυπτογράφηση. Η παραγωγή υπογραφής και πιστοποιητικών συγκριτικά με τα σχήματα [20] [21], είναι αρκετά πιο γρήγορη με τη χρήση των επιλεγμένων σχημάτων με τη χρήση της ECC. Συγκριτικά με το σχήμα ALI [25], η παρούσα υλοποίηση με ECC του σχήματος [28] ενδείκνυται να είναι πιο γρήγορη κατά την παραγωγή του μηνύματος που αφορά την επικοινωνία V2V, το VEHICLE_INFORM, αφού χρησιμοποιείται απλώς κρυπτογράφηση AES και ECDSA υπογραφή. Η HECC από την άλλη, καθυστερεί αρκετά χρονικά συγκριτικά με τις άλλες υλοποιήσεις.

Αυτό πιθανότατα οφείλεται στο γεγονός πως η επιλογή της καμπύλης, οι παράμετροι και η υλοποίηση της αριθμητικής παίζουν σημαντικό ρόλο στην επίδοση των κρυπτογραφικών τεχνικών. Επομένως, είναι ιδιαίτερα δύσκολη η υλοποίηση ενός εφαρμόσιμου κρυπτογραφικού συστήματος βασισμένο στην HECC με επιλογή παραμέτρων χειροκίνητα. Χρειάζεται προσεκτική ανάλυση και αρκετές δοκιμές, ενώ περισσότερη έρευνα στη χρήση τους μπορεί να ενισχύσει την επίδοσή τους. Αντίθετα, η ECC έχει ήδη διερευνηθεί σε μεγάλο βαθμό και έχουν αναπτυχθεί βιβλιοθήκες και βάσεις δεδομένων, οι οποίες διευκολύνουν τη χρήση τους χωρίς να είναι αναγκαίο να μελετηθεί σε βάθος το μαθηματικό τους υπόβαθρο. Επιπλέον, οι υλοποιήσεις είναι βελτιστοποιημένες με βάση την σύγχρονη έρευνα, η οποία συνεχώς βελτιώνει την επίδοση τους.

Από την άλλη, τα μεγέθη των μηνυμάτων είναι σχεδόν ίδια στις υλοποιήσεις με βάση την ECC και την HECC genus 2, πράγμα που είναι αναμενόμενο και μαθηματικά. Η χρήση της συμπίεσης μειώνει ιδιαίτερα τα μεγέθη των μηνυμάτων, ωστόσο τα μεγέθη για ασφάλεια του ίδιου επιπέδου αναμένεται να είναι ίδια σε περίπτωση που χρησιμοποιηθεί. Στην HECC genus 3 η συμπίεση παρουσιάζει δυσκολία στην μαθηματική θεμελίωσή της και δεν υπάρχει κάποιος διαδεδομένος αλγόριθμος, ο οποίος μπορεί να μεταφραστεί σε κώδικα με ευκολία. Αυτό δυσχεραίνει τη χρήση της στα δίκτυα των VANETs, αφού συγκριτικά με τους άλλους δύο αλγόριθμους δημιουργεί μεγαλύτερα μηνύματα, αλλά και μεγαλύτερη καθυστέρηση. Ωστόσο, και οι 3 αλγόριθμοι παράγουν σημαντικά μικρότερα μηνύματα, απ' ότι θα παρήγαγαν άλλοι αλγόριθμοι στο ίδιο επίπεδο ασφαλείας, τα οποία θα κατέκλυζαν το δίκτυο με πακέτα και με δικτυακές καθυστερήσεις στην επικοινωνία. Για παράδειγμα, το σχήμα [28], παρόλο που πετυχαίνει καλούς χρόνους με τη χρήση RSA, τα μηνύματα που προκύπτουν στο ίδιο επίπεδο ασφαλείας θα είναι σημαντικά πιο μεγάλα και δημιουργούν καθυστερήσεις στο δίκτυο. Στο σχήμα PPAAS [24] το μέγεθος του κρυπτογραφημένου μηνύματος είναι περίπου 212 bytes, ενώ στο σχήμα ALI [25], το οποίο αξιοποιεί ECC για τη μετάδοση των μηνυμάτων, ένα μήνυμα V2V έχει περίπου μέγεθος 149 bytes. Στην παρούσα υλοποίηση τα αντίστοιχα μηνύματα (VEHICLE_INFORM) είναι παρόμοιου μεγέθους με το ALI [25] και στις 3 υλοποιήσεις.

Όσον αφορά την κατανάλωση ενέργειας οι υλοποιήσεις δεν διέφεραν σημαντικά. Ωστόσο, είναι η μοναδική μετρική στην οποία υπερισχύουν οι αλγόριθμοι HECC σε σχέση με την ECC. Σε γενικές γραμμές, εφόσον τα μηνύματα είναι μικρά και δεν χρήζουν κατάτμησης η επικοινωνία δεν επιβαρύνει ενεργειακά τη συσκευή, πράγμα που είναι ιδιαίτερα επιθυμητό σε ενσωματωμένα συστήματα. Οι ενδείξεις είναι ενθαρρυντικές για τη χρήση της HECC σε τέτοιου είδους συστήματα, αφού από τα διαγράμματα συμπεραίνουμε ότι πιθανότατα να μπορεί να πετύχει μικρότερες καταναλώσεις.

Τελικά, συμπεραίνεται πως η κρυπτογραφία HECC είναι ακόμη σε πρώιμο στάδιο και πολλές από τις παραμέτρους για την υλοποίηση της χρειάζεται ιδιαίτερη προσοχή, ώστε να μπορεί να είναι εφαρμόσιμη σε πραγματικά σενάρια. Από την άλλη, οι ενδείξεις οδηγούν στην αισιοδοξία για τη χρήση τους στα VANETs. Με την κατάλληλη υλοποίηση και επιλογή παραμέτρων μπορεί να καταφέρει να πετύχει καλύτερες επιδόσεις από την ECC και να μειώσει τον υπολογιστικό φόρτο και την κατανάλωση ενέργειας των οχημάτων. Προφανώς, η χρήση της ECC είναι ιδιαίτερα συμφέρουσα σε τέτοιου τύπου δίκτυα, καθώς μειώνει σημαντικά τα μεγέθη των μηνυμάτων και την κατανάλωση ενέργειας. Η HECC με περαιτέρω βελτιώσεις στους υπολογισμούς και στις επιλογές των παραμέτρων μπορεί να την αντικαταστήσει με το πλεονέκτημα των μικρότερων κλειδιών και της καλύτερης επίδοσης στην ενέργεια και στον χρόνο.

6.3 Μελλοντικές επεκτάσεις

Οι μελλοντικές επεκτάσεις που προτείνονται βασίζονται κυρίως στη βελτίωση της επίδοσης της HECC. Αρχικά, η αριθμητική τους μπορεί να βελτιωθεί με χρήση πιο σύγχρονης βιβλιοθήκης για την αριθμητική υπολοίπου (modular arithmetic). Η χρήση της NTL 5.5, η οποία είναι ξεπερασμένη, ίσως δημιουργεί καθυστερήσεις στον βαθμωτό πολλαπλασιασμό. Επιπλέον, μπορεί να μελετηθεί η χρήση ειδικών καμπυλών (special curves [49]), στις οποίες έχουν προταθεί βελτιώσεις στην αριθμητική τους. Συμπεραίνεται, επίσης, πως η καλύτερη επιλογή των παραμέτρων μπορεί να βελτιώσει σημαντικά την επίδοση της HECC, όπως αναφέρεται στη δημοσίευση των Pelzl, Wollinger κ.α. [50]. Εκτός αυτών, για ενσωματωμένα συστήματα, μπορούν να χρησιμοποιηθούν και επιταχυντές υλικού σε πραγματικά συστήματα για τη βελτίωση της επίδοσής τους.

Επιπλέον, η επιλογή υπερελλειπτικών καμπυλών στο συγκεκριμένο σχήμα περιορίζεται από την έλλειψη ενός γενικού αλγόριθμου κωδικοποίησης μηνύματος στην καμπύλη. Ο αλγόριθμος [46] που χρησιμοποιήθηκε είναι ο πιο γενικευμένος αλγόριθμος που υπάρχει στη βιβλιογραφία και περιορίζει την επιλογή της καμπύλης σε συγκεκριμένες οικογένειες καμπυλών. Είναι σημαντικό να μελετηθεί ποιες από τις καμπύλες της οικογένειας είναι κρυπτογραφικά εφαρμόσιμες, δηλαδή παράγουν Ομάδα τάξης «σχεδόν» πρώτου αριθμού. Γι' αυτό μπορούν να χρησιμοποιηθούν αλγόριθμοι καταμέτρησης σημείων στην Ιακωβιανή Υπερελλειπτικών καμπυλών, ώστε να διαπιστωθεί ποιες από αυτές είναι επαρκώς ασφαλείς. Επιπλέον, η παραγωγή ασφαλών καμπυλών με τη χρήση της μεθόδου CM θα δημιουργήσει περισσότερες επιλογές ασφαλών καμπυλών με γνωστή τάξη, ώστε να χρησιμοποιηθούν καμπύλες ίδιου επιπέδου ασφαλείας και στις υπογραφές για καλύτερη μελέτη και σύγκριση.

Όσον αφορά την προσομοίωση, μπορεί να επεκταθεί, ώστε να συμπεριλάβει περισσότερα πραγματικά σενάρια και μεγαλύτερο φόρτο οχημάτων για καλύτερη αξιολόγηση. Μπορεί να ενσωματωθεί και τεχνική για την ανταλλαγή ψευδωνύμων και ανανέωσης κλειδιών για την μεγαλύτερη εμβάθυνση της χρήσης των κρυπτογραφικών τεχνικών που ερευνήθηκαν. Επιπλέον, μπορεί να διερευνηθεί μία καλύτερη λύση στο πρόβλημα που προέκυψε με την απόδειξη αρχηγίας, καθώς οι αλγόριθμοι ECC και HECC δεν έχουν τη δυνατότητα να κρυπτογραφήσουν ένα μήνυμα με το μυστικό κλειδί και στο συγκεκριμένο σχήμα μπορεί να δημιουργηθεί κενό ασφαλείας (να εντοπιστεί ο GL από έναν εξωτερικό επιτιθέμενο). Επιπλέον, μπορούν να ενσωματωθούν και κακόβουλοι χρήστες στην προσομοίωση, οι οποίοι θα επιτίθενται στο σχήμα ασφαλείας και θα πρέπει να εντοπίζονται από το σχήμα και να εξουδετερώνονται.

Κεφάλαιο 7: Επίλογος

Η παρούσα εργασία επικεντρώθηκε στην αξιολόγηση της χρήσης της κρυπτογραφίας HECC, συγκριτικά με αυτή της ECC, σε περιβάλλοντα VANETs. Αρχικά, αναλύθηκε το θεωρητικό υπόβαθρο, ώστε να γίνει καλύτερα κατανοητή η χρήση των κρυπτογραφικών τεχνικών, των ασφαλών σχημάτων και των δικτυακών ρυθμίσεων κατά την υλοποίηση. Στη συνέχεια, παρουσιάστηκε η σύγχρονη βιβλιογραφία και έγινε σύγκριση των διαφορετικών σχημάτων. Σε αυτή βασίστηκε η εργασία για να εξεταστούν νέες μέθοδοι με σκοπό τη βελτίωσή της. Αφότου αναλύθηκε η θεωρία και η βιβλιογραφία παρουσιάστηκε το πρόβλημα, το οποίο η εργασία προσπαθεί να αντιμετωπίσει.

Έπειτα, όσον αφορά την υλοποίηση, παρουσιάστηκαν τα εργαλεία, οι αλγόριθμοι και οι μετρικές που επιλέχθηκαν να την πλαισιώσουν, καθώς και οι λόγοι της επιλογής. Ακολούθησε η παρουσίαση της υλοποίησης της προσομοίωσης του ασφαλούς σχήματος στον δικτυακό προσομοιωτή NS-3 με τη χρήση των κρυπτογραφικών τεχνικών ECC, HECC genus 2 και 3. Τέλος, παρουσιάστηκαν τα αποτελέσματα της προσομοίωσης και σχολιάστηκαν τα συμπεράσματα, τα οποία προέκυψαν. Οι υλοποιήσεις των HECC δεν κατάφεραν να αντικρούσουν αυτή της ECC, ωστόσο, με τη χρήση των μεθόδων που αναφέρθηκαν στις μελλοντικές επεκτάσεις η κρυπτογραφία HECC μπορεί να αποτελέσει υποψήφια μέθοδος ενίσχυσης της ασφάλειας των VANETs.

Κεφάλαιο 8: Βιβλιογραφία

[1] S. S. Ghosh, H. Parmar, P. Shah and K. Samdani, «A Comprehensive Analysis Between Popular Symmetric Encryption Algorithms,» *2018 IEEE Punecon, Pune, India,* 2018.

[2] "What Is Asymmetric Encryption & How Does It Work," https://sectigostore.com/blog/what-is-asymmetric-encryption-how-does-it-work/. [Online]. [Accessed June 2023].

[3] "Diffie–Hellman key exchange," [Online]. Available: https://en.wikipedia.org/wiki/Diffie–Hellman_key_exchange. [Accessed June 2023].

[4] "Discrete logarithm," [Online]. Available: https://en.wikipedia.org/wiki/Discrete_logarithm. [Accessed June 2023].

[5] R. Kaur and A. Kaur, "Digital Signature," *2012 International Conference on Computing Sciences, Phagwara, India,* pp. 295-301, 2012.

[6] "Man-in-the-middle attack," [Online]. Available: https://en.wikipedia.org/wiki/Man-in-the-middle_attack. [Accessed June 2023].

[7] "Public key certificate," [Online]. Available: https://en.wikipedia.org/wiki/Public_key_certificate. [Accessed June 2023].

[8] "Group theory," [Online]. Available: https://en.wikipedia.org/wiki/Group_theory. [Accessed June 2023].

[9] "Cyclic group," [Online]. Available: https://en.wikipedia.org/wiki/Cyclic_group. [Accessed June 2023].

[10] "Finite field," [Online]. Available: https://en.wikipedia.org/wiki/Finite_field. [Accessed June 2023].

[11] R. Scheidler, "An Introduction to Hyperelliptic Curve Arithmetic".

[12] R. Alimoradi, "A Study of Hyperelliptic Curves in Cryptography," *I. J. Computer Network and Information Security,* pp. 67-72, 2016.

[13] "Wi-Fi," [Online]. Available: https://en.wikipedia.org/wiki/Wi-Fi. [Accessed June 2023].

[14] "802.11 Frame Types and Formats," [Online]. Available: https://howiwifi.com/2020/07/13/802-11-frame-types-and-formats/. [Accessed June 2023].

[15] «CWAP – HT Control Field,» [Ηλεκτρονικό]. Available: https://mrncciew.com/2014/10/20/cwap-ht-control-field/. [Πρόσβαση June 2023].

[16] Illa Ul Rasool, Yousaf Bin Zikria and Sung Won Kim, "A review of wireless access vehicular environment multichannel operational medium access control protocols: Quality-of-service analysis and other related issues," *International Journal of Distributed Sensor Networks,* 2017.

[17] R. G. Engoulou, M. Bellaïche, S. Pierre, A. Quintero, "VANET security surveys," *Computer Communications,* vol. 44, pp. 1-13, 2014.

[18] Mohammed Ali Hezam Al Junaid, Syed A. A et. al., «Classification of Security Attacks in VANET: A Review of Requirements and Perspectives,» *MATEC Web of Conferences,* 2018.

[19] "Elliptic-curve cryptography," [Online]. Available: https://en.wikipedia.org/wiki/Elliptic-curve_cryptography. [Accessed June 2023].

[20] Maria Azees, Pandi Vijayakumar, and Lazarus Jegatha Deboarh, "EAAP: Efficient Anonymous Authentication With Conditional Privacy-Preserving Scheme for Vehicular Ad Hoc Networks," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* vol. 18, no. 9, 2017.

[21] Yanbing Liu, Yuhang Wang, and Guanghui Chang, "Efficient Privacy-Preserving Dual Authentication and Key Agreement Scheme for Secure V2V Communications in an IoV Paradigm," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* vol. 18, no. 10, 2017.

[22] Hyo Jin Jo, In Seok Kim, and Dong Hoon Lee, "Reliable Cooperative Authentication for Vehicular Networks," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* vol. 19, no. 4, 2018.

[23] Hassan Mistareehi, Tariqul Islam, Kiho Lim & D. Manivannan, "A Secure and Distributed Architecture for Vehicular Cloud," *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing,* 2021.

[24] Yafang Yang, Lei Zhang et. al., "Privacy-Preserving Aggregation-Authentication Scheme for Safety Warning System in Fog-Cloud Based VANET," *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY,* vol. 17, 2022.

[25] Mir Ali Rezazadeh Baee, Leonie Simpson, Xavier Boyen, Ernest Foo and Josef Pieprzyk, "ALI: Anonymous Lightweight Inter-Vehicle Broadcast Authentication with Encryption," *TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING,* vol. X, no. X, 2021.

[26] Moumena Chaqfeh, Nader Mohamed, Imad Jawhar and Jie Wu, "Vehicular Cloud Data Collection for Intelligent Transportation Systems," 2016.

[27] Amit Dua, Neeraj Kumar, Ashok Kumar Das and Willy Susilo, "Secure Message Communication Protocol Among Vehicles in Smart City," *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY,* vol. 67, no. 5, 2018.

[28] Hassan Mistareehi and D. Manivannan, "A Low-Overhead Message Authentication and Secure Message Dissemination Scheme for VANETs," *Network,* vol. 2, pp. 139-152, 2022.

[29] "What is ns-3," [Online]. Available: https://www.nsnam.org/about/what-is-ns-3/. [Accessed June 2023].

[30] Michael Behrisch, Laura Bieker, Jakob Erdmann, Daniel Krajzewicz, "SUMO – Simulation of Urban MObility: An Overview," *SIMUL 2011,* 2011.

[31] Microchip, "RSA vs. ECC Comparison for Embedded Systems," [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/00003442A.pdf.

[32] "Advanced Encryption Standard," [Online]. Available: https://en.wikipedia.org/wiki/Advanced_Encryption_Standard. [Accessed June 2023].

[33] "ElGamal encryption," [Online]. Available: https://en.wikipedia.org/wiki/ElGamal_encryption. [Accessed June 2023].

[34] "Elliptic Curve Digital Signature Algorithm," [Online]. Available: https://en.wikipedia.org/wiki/Elliptic_Curve_Digital_Signature_Algorithm. [Accessed June 2023].

[35] C. Research, "SEC 4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV)," *Standards for Efficient Cryptography,* 2013.

[36] "Advanced Encryption Standard," [Online]. Available: https://www.cryptopp.com/wiki/Advanced_Encryption_Standard. [Accessed June 2023].

[37] "Standard curve database: secp256r1," [Online]. Available: https://neuromancer.sk/std/secg/secp256r1.

[38] "Elliptic Curve Cryptography," [Online]. Available: https://www.cryptopp.com/wiki/Elliptic_Curve_Cryptography. [Accessed June 2023].

[39] Padma Bh, D. Chandravathi, P. Prapoorna Roja, "Encoding And Decoding of a Message in the Implementation of Elliptic Curve Cryptography using Koblitz's

Method," *International Journal on Computer Science and Engineering,* vol. 02, no. 05, pp. 1904-1907, 2010.

[40] "Quadratic residue," [Online]. Available: https://en.wikipedia.org/wiki/Quadratic_residue. [Accessed June 2023].

[41] "NTL: A Library for doing Number Theory," [Online]. Available: https://libntl.org. [Accessed June 2023].

[42] "Elliptic Curve Digital Signature Algorithm," [Online]. Available: https://www.cryptopp.com/wiki/Elliptic_Curve_Digital_Signature_Algorithm. [Accessed June 2023].

[43] "libg2hec," [Online]. Available: https://github.com/syncom/libg2hec/tree/master. [Accessed June 2023].

[44] HENRI COHEN and GERHARD FREY, Handbook of Elliptic and Hyperelliptic Curve Cryptography, Chapman & Hall/CRC Taylor & Francis Group, 2006.

[45] Pierrick Gaudry & Éric Schost, "Construction of Secure Random Curves of Genus 2 over Prime Fields," *Cachin, C., Camenisch, J.L. (eds) Advances in Cryptology - EUROCRYPT,* vol. 3027, pp. 239-256, 2004.

[46] Michel Seck & Nafissatou Diarra , "Unified Formulas for Some Deterministic Almost-Injective Encodings into Hyperelliptic Curves," *Joux, A., Nitaj, A., Rachidi, T. (eds) Progress in Cryptology – AFRICACRYPT,* vol. 10831, 2018.

[47] A. Weng, "A class of hyperelliptic CM-curves of genus three," 2001.

[48] "NS3-HelperScripts," [Online]. Available: https://github.com/addola/NS3-HelperScripts/tree/master. [Accessed June 2023].

[49] Jan Pelzl, Thomas Wollinger, Christof Paar, "Special Hyperelliptic Curve Cryptosystems of Genus Two: Efficient Arithmetic and Fast Implementation," *Embedded Cryptographic Hardware: Design and Security,* 2004.

[50] Jan Pelzl, Thomas Wollinger, Jorge Guajardo, and Christof Paar, "Hyperelliptic Curve Cryptosystems: Closing the Performance Gap to Elliptic Curves," *C.D. Walter et al. (Eds.): CHES,* pp. 351-365, 2003.

Παράρτημα

```
/* Class for encoding integers in HEC of genus 1-3. Field should
be of
characteristic p = 7 mod 8 and p = 3 mod 4. */
class UnifiedEncoding {
    private:
        int inu, inw, ing;
        ZZ p, g, alpha_g, beta_g, gamma_g, mg, ng;
        ZZ_p u, w, s;
        poly_t fpoly1;
        NS_G2_NAMESPACE::g2hcurve curve2;
        g3HEC::g3hcurve curve3;
    public:
        UnifiedEncoding(ZZ p, int u, int w, int g=2, ZZ_p s =
ZZ_p::zero());
        void checkParams();
        void checkSParam();
        int isquadratic(ZZ_p a);
        void create_curve();
        int encode(ZZ val, ZZ_p &x, ZZ_p &y);
        int decode(ZZ_p& val1, ZZ_p& val2, ZZ_p x, ZZ_p y);
        NS_G2_NAMESPACE::g2hcurve getcurve();
        g3HEC::g3hcurve getcurveg3();
};

UnifiedEncoding::UnifiedEncoding(ZZ p, int u, int w, int g, ZZ_p
s) {
    this->inu = u;
    this->inw = w;
    this->ing = g;
    this->p = p;
    field_t::init(p);
    this->u = to_ZZ_p(to_ZZ(u));
    this->w = to_ZZ_p(to_ZZ(w));
    this->s = s;
    this->g = to_ZZ(g);
    this->alpha_g = to_ZZ(pow(2, (2*g - 1)) - 1);
    this->beta_g = 4*g*g + 2*g;
    this->gamma_g = pow((2*g*g+g), 2);
    if(this->g%2 == 0) {
        this->mg = (this->alpha_g*this->beta_g)/4;
    }
    else {
        this->mg = (this->alpha_g*this->beta_g)/2;
    }
```

```
    if(this->g%2 == 0) {
        this->ng = pow((2*g*g+g),2)/2;
    }
    else {
        this->ng = pow((2*g*g+g),2);
    }
    checkParams();
    checkSParam();
    create_curve();
}

void UnifiedEncoding::checkParams() {
    if(p%2 == 0 || (2*g*g +g)%p == 0) {
        std::cout << "Error:\n\tWrong input for field
characteristic p!" << std::endl;
        exit(1);
    }
    if(p%8 != 7) {
        std::cout << "Error:\n\tField q is not 7 modulo 8!" <<
std::endl;
        exit(1);
    }
    if(w==0 || u==0) {
        std::cout << "Error:\n\tu or w is zero!" << std::endl;
        exit(1);
    }
    ZZ_p check = squareRoot(u, p);
    if(check != 0) {
        std::cout << "Error:\n\tu is a square!" << std::endl;
        exit(1);
    }
    if(g >= 6){
        std::cout << "Error:\n\tGenus out of range!" <<
std::endl;
        exit(1);
    }
}

void UnifiedEncoding::checkSParam() {
    if (s==0) {
        if(alpha_g%p == 0)
            s = to_ZZ_p(gamma_g/beta_g);
        else {
            ZZ_p delta_s = to_ZZ_p(beta_g*beta_g +
4*alpha_g*gamma_g);
            s = (-to_ZZ_p(beta_g)+squareRoot(delta_s,
p))/(to_ZZ_p(2*alpha_g));
```

```
        }
    }
}

int UnifiedEncoding::isquadratic(ZZ_p a){
    if(a == 0)
        return 0;
    else if(squareRoot(a, p) != 0)
        return 1;
    else
        return -1;
}

void UnifiedEncoding::create_curve() {
    ZZ_p a0, a2g, a1, a3;
    a0 = (s-(to_ZZ_p(2*g*g +g)))/(to_ZZ_p(2*g*g+g));
    a0 = a0*pow(inw, 2*ing+1);
    a2g = s*w*w;
    a1 = (s*pow(inw, 2*ing))/to_ZZ_p(g);
    SetCoeff(fpoly1, 0, a0);
    SetCoeff(fpoly1, 1, a1);
    SetCoeff(fpoly1, 2*ing-1, a2g);
    if(ing == 3) {
        a3 = ((2*ing-1)*s*w*w*w*w)/3;
        SetCoeff(fpoly1, 3, a3);
    }
    SetCoeff(fpoly1, 2*ing+1, 1);
    if(ing == 2) {
        curve2.set_f(fpoly1);
        curve2.update();
    }
    if(ing == 3) {
        curve3.set_f(fpoly1);
        curve3.update();
    }
}

int UnifiedEncoding::encode(ZZ val, ZZ_p &x, ZZ_p &y) {
    ZZ_p r = to_ZZ_p(val);
    ZZ_p check = eval(fpoly1, r);
    if(check == 0) {
        std::cout << "Value: " << r << "is not in the supported
range, maybe increase by 1." << std::endl;
        return 1;
    }
    ZZ_p v = w*(u*r*r*(to_ZZ_p(-mg)*s+to_ZZ_p(-ng)) + to_ZZ_p(-
1));
    int e = isquadratic(eval(fpoly1, v));
```

```
    ZZ_p x1 = to_ZZ_p((1+e)/2)*v + to_ZZ_p((1-e)/2)*(w*(-
v+w)/(v+w));
    y = to_ZZ_p(-e)*squareRoot(eval(fpoly1, x1),p);
    x = x1;
    return 0;
}

int UnifiedEncoding::decode(ZZ_p &val1, ZZ_p &val2, ZZ_p x, ZZ_p
y){

    if(eval(fpoly1, x) != y*y) {
        std::cout << "Point given is not a point of the
hyperelliptic curve!" << std::endl;
        return 1;
    }
    if(isquadratic(u*w*(x+w)*(to_ZZ_p(-ng) + to_ZZ_p(-mg)*s)) !=
1) {
        std::cout << "u*w*(x+w)*(-ng-mg*s) is not a square in Fq"
<< std::endl;
        return 1;
    }
    ZZ_p hlp = u*w*(to_ZZ_p(-ng)+ to_ZZ_p(-mg)*s);
    ZZ_p hlp2 = u*(x+w)*(to_ZZ_p(-ng) + to_ZZ_p(-mg)*s);
    val1 = squareRoot((x+w)/hlp, p);
    val2 = squareRoot(2*w/hlp2, p);

    return 0;
}
```

```
NS_G2_NAMESPACE::divisor points_to_divisor (ZZ_p x1, ZZ_p y1,
ZZ_p x2, ZZ_p y2, NS_G2_NAMESPACE::g2hcurve curve) {
    NS_G2_NAMESPACE::divisor D;
    ZZ_p a = -x1 -x2;
    ZZ_p b = x1*x2;
    poly_t u,v;
    SetCoeff(u, 2, 1);
    SetCoeff(u, 1, a);
    SetCoeff(u, 0, b);

    ZZ_p c = (y1-y2)/(x1-x2);
    ZZ_p d = y1 - c*x1;
    SetCoeff(v, 1, c);
    SetCoeff(v, 0, d);
```

```
    D.set_curve(curve);
    D.set_upoly(u);
    D.set_vpoly(v);
    D.update();
    return D;
}

g3HEC::g3divisor points_to_divisorg3 (ZZ_p x1, ZZ_p y1, ZZ_p x2,
ZZ_p y2, ZZ_p x3, ZZ_p y3, g3HEC::g3hcurve curveg3) {
    g3HEC::g3divisor D;
    ZZ_p a,b,c;
    a = -x1-x2-x3;
    b = x1*x2 + x1*x3 + x2*x3;
    c = -x1*x2*x3;

    poly_t u,v;
    SetCoeff(u, 3, 1);
    SetCoeff(u, 2, a);
    SetCoeff(u, 1, b);
    SetCoeff(u, 0, c);

    ZZ_p e = ((y2-y3)*(x1*x1 - x2*x2) - (y1-y2)*(x2*x2 -
x3*x3))/((x2-x3)*(x1*x1 - x2*x2) - (x1-x2)*(x2*x2 - x3*x3));
    ZZ_p d = (y1 - y2 - e*(x1-x2))/(x1*x1 - x2*x2);
    ZZ_p f = y3 - e*x3 - d*x3*x3;
    SetCoeff(v, 2, d);
    SetCoeff(v, 1, e);
    SetCoeff(v, 0, f);

    D.set_curve(curveg3);
    D.set_upoly(u);
    D.set_vpoly(v);
    D.update();
    return D;
}

void divisor_to_points (NS_G2_NAMESPACE::divisor D, ZZ_p &x1,
ZZ_p &y1, ZZ_p &x2, ZZ_p &y2, ZZ p) {
  poly_t u,v;
  u = D.get_upoly();
  v = D.get_vpoly();

  ZZ_p a,b,c,d;
  GetCoeff(a,u,1);
  GetCoeff(d,v,0);
  GetCoeff(c,v,1);

  if(DetIrredTest(u)){
```

```
    std::cout << "Invalid divisor of genus 2 for converting to
text" << std::endl;
    exit(1);
  }
  x1 = FindRoot(u);
  x2 = -x1 - a;

  y1 = d + c*x1;
  y2 = y1 - c*(x1-x2);
}

void divisorg3_to_points(g3HEC::g3divisor D, ZZ_p &x1, ZZ_p &y1,
ZZ_p &x2, ZZ_p &y2, ZZ_p &x3, ZZ_p &y3, ZZ p){
    poly_t u, v;
    u = D.get_upoly();
    v = D.get_vpoly();

    if(DetIrredTest(u)) {
        std::cout << "Invalid divisor of genus 3 for converting
to text" << std::endl;
        exit(1);
    }
    vec_ZZ_p roots = FindRoots(u);
    x1 = roots[0];
    x2 = roots[1];
    x3 = roots[2];

    ZZ_p d,e,f;
    y1 = eval(v, x1);
    y2 = eval(v, x2);
    y3 = eval(v, x3);
}
```

```
int divisor_to_bytes(uint8_t *buff, NS_G2_NAMESPACE::divisor D,
NS_G2_NAMESPACE::g2hcurve curve, ZZ p) {
    int size = NTL::NumBytes(p);
    poly_t u = D.get_upoly();
    poly_t v = D.get_vpoly();
    poly_t f = curve.get_f();
    ZZ_p c1, c2;
    GetCoeff(c1, u, 1);
    GetCoeff(c2, u, 0);
    uint8_t *c1z, *c2z;
    c1z = new uint8_t[size];
    NTL::BytesFromZZ(c1z, rep(c1), size);
    c2z = new uint8_t[size];
```

```
    NTL::BytesFromZZ(c2z, rep(c2), size);
    memcpy(buff, c1z, size);
    memcpy(buff+size, c2z, size);
    ZZ_p s0, v0, f0;
    GetCoeff(v0, v, 0);
    GetCoeff(f0, f, 0);

    ZZ_p f1, f2, f3, f4, v1;
    GetCoeff(v1, v, 1);
    GetCoeff(f1, f, 1);
    GetCoeff(f2, f, 2);
    GetCoeff(f3, f, 3);
    GetCoeff(f4, f, 4);
    if(c2 != 0)
        s0 = (v0*v0-f0)/c2;
    else {

        s0 = v1*v1 - f2 +f3*c1 + f4*(c2 - c1*c1) - c1*(2*c2 -
c1*c1);
    }
    if((c1*c1 - 4*c2) != 0) {
        poly_t bsq, ap, gp;
        SetX(bsq);
        SetCoeff(bsq, 1, c1);
        SetCoeff(bsq, 0, (f1 - f3*c2 + f4*c2*c1 + c2*(c2 -
c1*c1)));
        bsq = bsq*bsq;

        SetX(ap);
        SetCoeff(ap, 0, (f2 - f3*c1 - f4*(c2 - c1*c1) + c1*(2*c2
- c1*c1)));
        SetX(gp);
        SetCoeff(gp, 1, c2);
        SetCoeff(gp, 0, f0);

        poly_t ds0 = bsq - 4*ap*gp;
        MakeMonic(ds0);
        vec_ZZ_p roots = FindRoots(ds0);
        if((s0 == roots[0]) && (rep(roots[0]) < rep(roots[1])))
            buff[2*size] = 0;
        else if ((s0 == roots[0]) && (rep(roots[0]) >
rep(roots[1])))
            buff[2*size] = 1;
        else if ((s0 == roots[1]) && (rep(roots[1]) >
rep(roots[0])))
            buff[2*size] = 1;
        else
            buff[2*size] = 0;
```

```
    }
    else {
        buff[2*size] = 0;
    }

    if(v0!=0){
        if(rep(v0)%2 != 0) {
            buff[2*size] += 2;
        }
    }
    else {
        if(rep(v1)%2 != 0) {
            buff[2*size] += 2;
        }
    }
    return 0;
}

int bytes_to_divisor(NS_G2_NAMESPACE::divisor &D, uint8_t *buff,
NS_G2_NAMESPACE::g2hcurve curve, ZZ p) {
    int size = NTL::NumBytes(p);
    ZZ c1, c2;
    c1 = NTL::ZZFromBytes(buff, size);
    c2 = NTL::ZZFromBytes(buff+size, size);
    poly_t u,v;
    ZZ_p u1 = to_ZZ_p(c1);
    ZZ_p u0 = to_ZZ_p(c2);
    SetCoeff(u,2,1);
    SetCoeff(u,1,u1);
    SetCoeff(u,0,u0);

    uint8_t bits = buff[2*size];
    poly_t f = curve.get_f();
    poly_t bsq, ap, gp;
    ZZ_p f0, f1, f2, f3, f4;
    GetCoeff(f0, f, 0);
    GetCoeff(f1, f, 1);
    GetCoeff(f2, f, 2);
    GetCoeff(f3, f, 3);
    GetCoeff(f4, f, 4);
    SetX(bsq);
    SetCoeff(bsq, 1, u1);
    SetCoeff(bsq, 0, (f1 - f3*u0 + f4*u0*u1 + u0*(u0 - u1*u1)));
    bsq = bsq*bsq;

    SetX(ap);
    SetCoeff(ap, 0, (f2 - f3*u1 - f4*(u0 - u1*u1) + u1*(2*u0 -
u1*u1)));
```

```
    SetX(gp);
    SetCoeff(gp, 1, u0);
    SetCoeff(gp, 0, f0);

    poly_t ds0 = bsq - 4*ap*gp;
    MakeMonic(ds0);
    if(DetIrredTest(ds0)) {
        return 1;
    }
    vec_ZZ_p roots = FindRoots(ds0);
    ZZ_p s0;
    if(roots[0] == roots[1])
        s0 = roots[0];
    else{
        if((bits & 1) == 1)
            s0 = (rep(roots[0]) < rep(roots[1])) ? roots[1] :
roots[0];
        else
            s0 = (rep(roots[0]) < rep(roots[1])) ? roots[0] :
roots[1];
    }

    ZZ_p v0sq, v0, v1;
    v0sq = u0*s0 + f0;
    if(v0sq != 0) {
        v0sq = squareRoot(v0sq, p);
        if((bits & 2) == 2){
            v0 = (rep(v0sq)%2 == 1) ? v0sq : -v0sq;
        }
        else
            v0 = (rep(v0sq)%2 == 1) ? -v0sq : v0sq;
        v1 = (u1*s0 + f1 - f3*u0 + f4*u0*u1 + u0*(u0 -
u1*u1))/(2*v0);
    }
    else{
        v0 = 0;
        ZZ_p v1sq;
        v1sq = s0 + f2 - f3*u1 - f4*(u0 - u1*u1) + u1*(2*u0 -
u1*u1);
        v1sq = squareRoot(v1sq, p);
        if((bits & 2) == 2){
            v1 = (rep(v1sq)%2 == 1) ? v1sq : -v1sq;
        }
        else
            v1 = (rep(v1sq)%2 == 1) ? -v1sq : v1sq;
    }

    SetCoeff(v, 1, v1);
```

```
    SetCoeff(v, 0, v0);
    D.set_curve(curve);
    D.set_upoly(u);
    D.set_vpoly(v);
    D.update();
    return 0;
}
```