

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΔΙΑΔΙΚΤΥΟ ΤΩΝ ΠΡΑΓΜΑΤΩΝ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΟ ΝΕΦΟΣ ΣΤΑ ΕΥΦΥΗ
ΟΧΗΜΑΤΑ

Διπλωματική Εργασία

του

Δημητρίου Λούβαρη

Θεσσαλονίκη, Οκτώβριος 2020

ΔΙΑΔΙΚΤΥΟ ΤΩΝ ΠΡΑΓΜΑΤΩΝ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΟ ΝΕΦΟΣ ΣΤΑ ΕΥΦΥΗ
ΟΧΗΜΑΤΑ

Δημήτριος Λούβαρης

Πτυχίο Μηχανικού Οχημάτων Τ.Ε, ΑΤΕΙΘ, 2012

Διπλωματική Εργασία

υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ
ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων Καθηγητής,
κ. Ελευθέριος Μαμάτας

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την

Όνοματεπώνυμο 1

Όνοματεπώνυμο 2

Όνοματεπώνυμο 3

.....

.....

.....

Δημήτριος Λούβαρης

.....

Περίληψη

Το Διαδίκτυο των Πραγμάτων (IoT) είναι μια τεχνολογία που βρίσκει απήχηση σε πολλούς τομείς σε παγκόσμιο επίπεδο, όπως έξυπνες πόλεις, ο κατασκευαστικός τομέας, η εφοδιαστική αλυσίδα και οι κρίσιμες υποδομές. Πάνω από το Διαδίκτυο των Πραγμάτων, οι αρχιτεκτονικές επιλογές του υπολογιστικού νέφους μετατοπίζονται από νέφο-κεντρικές σε ακρο-κεντρικές, εκφορτώνοντας όλο και περισσότερες λειτουργίες από το κεντρικό νέφος (Core Cloud) στις συσκευές Άκρης (Edge Cloud). Επομένως, οι συσκευές υπολογιστικής Άκρης γίνονται πιο σύνθετες παρέχοντας λειτουργίες επεξεργασίας δεδομένων και λήψης αποφάσεων, επιταχύνοντας έτσι την αποκέντρωση του νέφους.

Η παρούσα διπλωματική εργασία, εντάσσεται στον τομέα των ευφυών μεταφορών και οχημάτων και σκοπός της είναι η ανάπτυξη και η μελέτη της τεχνολογίας της υπολογιστικής των άκρων σε αυτά τα περιβάλλοντα, στοχεύοντας στην μείωση της καθυστέρησης μετάδοσης των δεδομένων αλλά και στη μείωση των απαιτούμενων πόρων από το κεντρικό νέφος. Αναπτύξαμε και αξιολογήσαμε την εφαρμογή μιας συσκευής άκρου (ως κόμβο υπολογιστικής άκρης) σε αυτοκίνητο, πραγματοποιώντας συλλογή δεδομένων από τις μετρήσεις των αισθητήρων του αυτοκινήτου, καταγράφοντας την καθυστέρηση μετάδοσης (latency) και την κατανάλωση του εύρους ζώνης (bandwidth).

Για την υλοποίηση της μελέτης χρησιμοποιήθηκε το Raspberry Pi ως συσκευή Edge, μια πλατφόρμα IoT ως εξυπηρετητής για τη μεταφόρτωση μηνυμάτων, ένα OBD2 reader για την λήψη δεδομένων από τους αισθητήρες του οχήματος και ένα smartphone ως 4G modem. Μελετώνται τα προβλήματα συλλογής δεδομένων από τους αισθητήρες του αυτοκινήτου, της άμεσης απόφασης σε περίπτωση βλάβης και της αποδοτικής επικοινωνίας των δεδομένων στο υπολογιστικό νέφος.

Μέσα από τα αποτελέσματα της πειραματικής μας ανάλυσης παρατηρήσαμε σημαντική μείωση της καθυστέρησης της μετάδοσης, παρουσιάζοντας μια βελτίωση περίπου 35% με την χρήση της υπολογιστικής των άκρων. Επίσης, όσον αφορά την κατανάλωση εύρους ζώνης, υπήρξε μια σχετική μείωση με την χρήση της συσκευής Edge λόγω της επεξεργασίας των δεδομένων πριν την αποστολή στο Cloud.

Λέξεις Κλειδιά: Διαδίκτυο των Πραγμάτων, Ευφυή οχήματα, OBD, Raspberry, Υπολογιστική Άκρου

Abstract

Internet of Things is a technology that resonates in many areas worldwide, such as smart cities, the construction sector, the supply chain and critical infrastructure. On top of Internet of Things, cloud computing architectural options are shifting from cloud-centric to edge-centric, offloading more and more cloud-based functions from Core Cloud to Edge Cloud devices. As a result, Edge computing devices become more complex by providing data processing and decision-making functions, accelerating cloud decentralization.

This dissertation is part of the field of intelligent transport and vehicles and its purpose is the development and study of edge computing technology in these environments, aiming to reduce data transmission delays and reduce the resources required by the Core Cloud. We developed and evaluated the application of an Edge device (as a computing edge node) in a car, collecting data from the vehicle's sensors, monitoring the latency and the bandwidth consumption.

The study used the Raspberry Pi as an edge device, an IoT platform as server, an OBD2 reader for vehicle's sensors data collection and a smartphone which acts as a 4G modem. The issues that studied are: data collection from the car's sensors, immediate decision in case of failure and efficient communication of the data in the cloud computing.

Through the results of our experimental analysis we observed a significant reduction in transmission delay, showing an improvement of about 35% with the use of Edge computing. Also in terms of bandwidth consumption, there was a relative reduction with the use of the Edge device due to the processing of data before sending to the Cloud.

Keywords: Internet of Things, Smart vehicles, OBD, Raspberry, Edge Computing

Περιεχόμενα

1	Εισαγωγή	1
1.1	Πρόβλημα – Σημαντικότητα του θέματος	1
1.2	Σκοπός – Στόχοι	2
1.3	Διάρθρωση της μελέτης	2
2	Το Διαδίκτυο των Πραγμάτων	3
2.1	Ιστορική ανασκόπηση	3
2.2	Ορισμός του IoT	4
2.3	Τεχνολογία του Διαδικτύου των Πραγμάτων	6
2.4	Μοντέλα επικοινωνίας στο Διαδίκτυο των Πραγμάτων	9
2.5	Το Διαδίκτυο των Οχημάτων	11
3	Υπολογιστικό νέφος (Cloud Computing)	14
3.1	Τι είναι το υπολογιστικό νέφος;	14
3.2	Υπηρεσίες και μοντέλα ανάπτυξης Υπολογιστικού Νέφους	15
3.3	Οφέλη χρήσης υπολογιστικού νέφους	17
3.4	Ασφάλεια στο υπολογιστικό νέφος	18
3.5	Το υπολογιστικό νέφος στα συνδεδεμένα οχήματα	18
3.6	Υπολογιστική Ομίχλης (Fog Computing)	20
3.7	Ενσωμάτωση Διαδικτύου των Πραγμάτων και Υπολογιστικής Άκρου	23
3.8	Πλεονεκτήματα Edge Computing	24
3.9	Ασφάλεια και ιδιωτικότητα στο Άκρο του δικτύου	25
3.10	Υπολογιστική Άκρου στα συνδεδεμένα οχήματα	26
4	Ανάπτυξη πειραματικού περιβάλλοντος	28
4.1	Τεχνολογίες Hardware	29
4.1.1	Raspberry Pi	30
4.1.2	Διεπαφή OBDII	30
4.1.3	Συσκευή διάγνωσης ELM327	33
4.1.4	Τεχνολογία Bluetooth	33
4.1.5	Ασύρματα τοπικά δίκτυα (Tethering, 4G, LTE)	34
4.1.6	Λειτουργία Tethering και δίκτυο 4G	35
4.2	Τεχνολογίες Software	36
4.2.1	Λειτουργικό σύστημα Raspberry Pi	36

4.2.2 Εφαρμογή Torque	36
4.2.3 Node-RED	37
4.2.4 Cloud Azure IoT	38
4.2.5 Docker	41
4.2.6 Πρωτόκολλο HTTP	41
4.3 Περιβάλλον πειράματος	43
4.4 Υλοποίηση πειραματικής προσέγγισης	44
4.4.1 Εξυπηρετητής Azure – Πελάτης Raspberry	45
4.4.2 Σύνδεση OBD2 reader με την εφαρμογή Torque	47
4.4.3 Μεταφόρτωση δεδομένων μέσα από την εφαρμογή Torque	49
4.4.4 Ανάπτυξη ροής στο Node-RED και ανάλυση των κόμβων	50
4.5 Παρακολούθηση απόδοσης δικτύου	53
5 Πειραματική ανάλυση	53
5.1 Πειραματική μεθοδολογία	54
5.2 Πειραματικά αποτελέσματα	56
6 Επίλογος	60
6.1 Σύνοψη και συμπεράσματα	60
6.2 Όρια και περιορισμοί της έρευνας	61
6.3 Μελλοντικές Επεκτάσεις	62
Βιβλιογραφία	63

Κατάλογος Εικόνων

Εικόνα 2.1-1: Εφαρμογές IoT (IoT Applications, edureka.co).....	3
Εικόνα 2.3-1: Αρχιτεκτονική IoT.....	7
Εικόνα 2.4-1: Μοντέλο επικοινωνίας μηχανή προς μηχανή.....	9
Εικόνα 2.4-2: Μοντέλο επικοινωνίας μηχανή προς νέφος.....	10
Εικόνα 2.4-3: Μοντέλο επικοινωνίας μηχανή προς πύλη.....	11
Εικόνα 2.5-1: Τύποι επικοινωνίας του οχήματος στο IoV.....	14
Εικόνα 3.2-1: Μοντέλα υπηρεσιών υπολογιστικού νέφους.....	16
Εικόνα 3.5-1: Χαρτογράφηση οδικού περιβάλλοντος από το αυτόνομο όχημα.....	19
Εικόνα 3.6-1: Χρήση Edge Computing.....	21
Εικόνα 3.6-2: Δομή Edge Computing (Front, Near, Far-End).....	22
Εικόνα 4.1-1: Θύρα OBD2 σε NISSAN NV300.....	31
Εικόνα 4.1-2: OBD2 Adapter ELM327.....	33
Εικόνα 4.2-1: Γραφικό περιβάλλον εφαρμογής Torque.....	37
Εικόνα 4.2-2: Γραφικό περιβάλλον σύνταξης ροών του Node-RED.....	38
Εικόνα 4.2-3: Γραφικό περιβάλλον Azure IoT Hub.....	41
Εικόνα 4.4-1: Εισαγωγή container στην συσκευή Edge του Azure.....	47
Εικόνα 4.4-2: Εισαγωγή διαδρομής μηνυμάτων.....	47
Εικόνα 4.4-3: Επικοινωνία ELM327 με Torque.....	48
Εικόνα 4.4-4: Πραγματικές τιμές αισθητήρων.....	49
Εικόνα 4.4-5: Ρυθμίσεις και επιλογές καταγραφής δεδομένων.....	50
Εικόνα 4.4-6: Ροή Node-RED της προτεινόμενης Edge αρχιτεκτονικής.....	51
Εικόνα 4.4-7: Ροή node-RED με λειτουργία άμεσης ειδοποίησης σφάλματος ενός αισθητήρα.....	52
Εικόνα 5.1-1: Μηνύματα τιμών αισθητήρων από φορητή υπολογιστική μονάδα προς το Azure IoT Hub.....	55
Εικόνα 5.1-2: Μηνύματα τιμών αισθητήρων από συσκευή Edge προς το Hub.....	56

Κατάλογος Πινάκων

Πίνακας 3.7-1: Χαρακτηριστικά IoT Edge Cloud Computing.....	23
Πίνακας 4.2-1: Περιγραφή μεθόδων HTTP.....	42
Πίνακας 4.3-1: Τεχνικά χαρακτηριστικά και εκδόσεις εργαλείων πειράματος.....	43
Πίνακας 5-1: Συγκριτική ανάλυση Cloud και Edge Computing.....	59

Κατάλογος Διαγραμμάτων

Διάγραμμα 4-1: Αρχιτεκτονική συστήματος Edge.....	45
Διάγραμμα 5-1: RTT από αισθητήρες προς Core Cloud.....	57
Διάγραμμα 5-2: I/O graph κατά την μεταφόρτωση από αισθητήρες προς Core Cloud....	57
Διάγραμμα 5-3: RTT από συσκευή Edge προς Core Cloud.....	58
Διάγραμμα 5-4: I/O graph κατά την μεταφόρτωση από συσκευή Edge προς Core Cloud.....	58

Συμβολισμοί

Συμβολισμός	Επεξήγηση
XML	eXtensible Markup Language
UDP	User Datagram Protocol
TCP	Transmission Control Protocol
IP	Internet Protocol
REST	Representational state transfer
HTTP	HyperText Transfer Protocol
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
MQTT	Message Queuing Telemetry Transport
WSN	Wireless sensor network
IoT	Internet of Things
PaaS	Platform as a Service
SaaS	Software as a Service
IaaS	Infrastructure as a Service
WLAN	Wireless Local Area Network
Wi-Fi	WirelessFidelity
SMTP	Simple Mail Transfer Protocol
RTT	RoundTripTime

1 Εισαγωγή

1.1 Πρόβλημα – Σημαντικότητα του θέματος

Τα τελευταία έτη με την εξέλιξη των Ασυρμάτων Δικτύων Αισθητήρων (Wireless Sensors Networks, WSNs) και των τεχνολογιών Διαδικτύου ολοένα και περισσότεροι χρήστες, υπηρεσίες και βιομηχανίες στράφηκαν στο Διαδίκτυο των Πραγμάτων. Επιπλέον με την μαζική μετάβαση σε υπηρεσίες Υπολογιστικού Νέφους, σε συνδυασμό με το Διαδίκτυο των Πραγμάτων, έχει ξεκινήσει η ανάπτυξη τεχνολογικών καινοτομιών που έχουν σημαντικό αντίκτυπο στην επιστήμη και εν γένει στην κοινωνία.

Στον τομέα της αυτοκίνησης οι βιομηχανίες έχουν εισαγάγει στα οχήματα αισθητήρες και ενεργοποιητές, ώστε να επιτυγχάνονται μεγαλύτερη επίδοση, καλύτερη διαχείριση του καυσίμου, λιγότεροι ρύποι στα καυσαέρια και ανίχνευση βλαβών – σφαλμάτων του κινητήρα μέσω του συστήματος Διαγνωστικός Έλεγχος Αυτοκινήτου (On-Board Diagnostics, OBD). Επιπρόσθετα οι εταιρείες αυτοκίνησης αξιοποιώντας την συνεχή εξέλιξη των πληροφοριακών συστημάτων, προσθέτουν περισσότερους αισθητήρες στα οχήματα όπως επιταχυνσιόμετρα, κάμερες κ.α. παράγοντας πιο «έξυπνα» οχήματα με στόχο να γίνουν πλήρως αυτόνομα. Χρησιμοποιώντας ένα «έξυπνο» όχημα αξιοποιείται το διαδίκτυο των πραγμάτων και η υπολογιστική νέφους για λόγους:

- Ασφάλειας σε περίπτωση ατυχήματος
- Ανάλυση των παραγομένων δεδομένων ούτως ώστε να βελτιωθούν οδικές υποδομές και ηλεκτρομηχανολογικά εξαρτήματα του οχήματος
- Προληπτικής συντήρησης
- Τηλεμετρίας και οδηγικής συμπεριφοράς
- Εφαρμογής μοντέλων πρόβλεψης ώστε να μειωθεί η λανθασμένη εκτίμηση του οδηγού – μείωση ατυχημάτων

Έτσι, αυξάνονται σε πολύ μεγάλο βαθμό τα οχήματα που συνδέονται στο IoT όπως επίσης αυξάνονται και τα παραγόμενα δεδομένα, επιβαρύνοντας τις υπηρεσίες Cloud, κατά την κίνηση των οχημάτων. Κατά την μεταφόρτωση των παραγόμενων δεδομένων σε κάποιο υπολογιστικό νέφος, το πρόβλημα που εντοπίζεται είναι ότι ο κεντρικός εξυπηρετητής βρίσκεται μακριά από το όχημα στο οποίο παράγονται τα δεδομένα, με αποτέλεσμα την αύξηση των πόρων που χρειάζονται για την μετάδοση, την μείωση του

εύρους ζώνης (bandwidth), την καθυστέρηση (latency) ή ακόμη και την διακοπή της αποστολής των δεδομένων λόγω μείωσης της έντασης του σήματος.

1.2 Σκοπός – Στόχοι

Ο σκοπός της διπλωματικής εργασίας είναι να κατανοηθεί πώς το IoT και οι τεχνολογίες Cloud μπορούν να συνεισφέρουν στη βελτίωση και την επίλυση βασικών προβλημάτων στον τομέα των οδικών μεταφορών.

Ο γενικός στόχος είναι η παρουσίαση της αποδοτικότητας και των δυνατοτήτων ενός Edge Cloud έναντι του κύριου νέφους κατά την δυναμική μεταφόρτωση δεδομένων από την θύρα OBD του οχήματος. Αναλυτικότερα, οι επιμέρους στόχοι καταμερίζονται ως εξής.

- Εφαρμογή και παρουσίαση εγκατάστασης Edge Cloud
- Καταγραφή και ανάλυση της καθυστέρησης κατά την μεταφόρτωση δεδομένων στο Edge Cloud και στο Cloud
- Λήψη άμεσης απόφασης του συστήματος σε περίπτωση βλάβης
- Αποδοτικής επικοινωνίας των δεδομένων στα υπολογιστικά νέφη σε περίπτωση μείωσης ή διακοπής του σήματος δικτύου

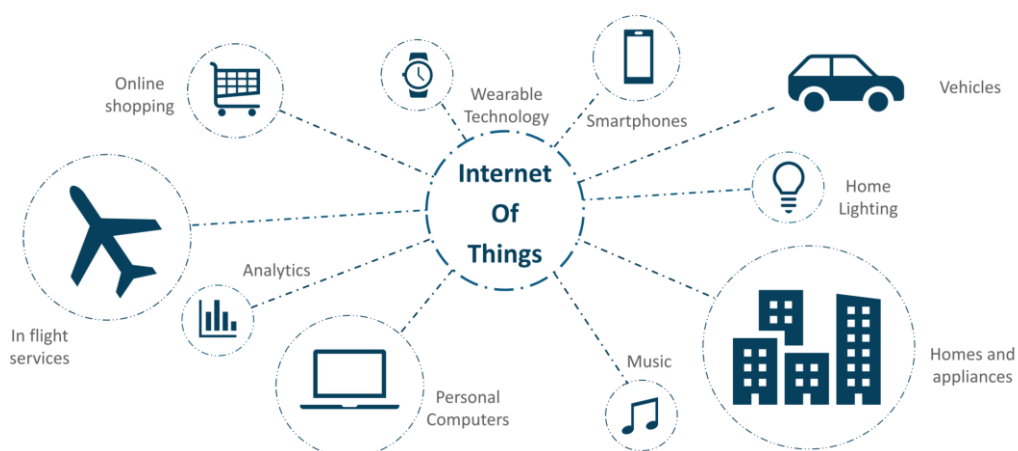
1.3 Διάρθρωση της μελέτης

Η διάρθρωση της εργασίας είναι η εξής: Στο **κεφάλαιο 2** γίνεται αναφορά στο θεωρητικό υπόβαθρο του Διαδικτύου των Πραγμάτων. Παρουσιάζεται η αρχιτεκτονική του, τα μοντέλα επικοινωνίας που χρησιμοποιεί και μια εισαγωγή στο Διαδίκτυο των Οχημάτων. Στο **κεφάλαιο 3** περιγράφεται η υπολογιστική νέφους και Άκρου. Περιλαμβάνονται μοντέλα ανάπτυξης, οφέλη και πλεονεκτήματα και αξιοποίηση τους από το πεδίο των οχημάτων. Στη συνέχεια στο **κεφάλαιο 4** παρουσιάζεται η ανάπτυξη του πειραματικού περιβάλλοντος και αναλύεται η υλοποίηση του. Στο **κεφάλαιο 5** εκτελούνται 2 πειράματα, λαμβάνονται μετρήσεις και παρουσιάζονται τα αποτελέσματα. Τέλος, στο **κεφάλαιο 6** όπου είναι και ο επίλογος της εργασίας, παρατίθενται τα συμπεράσματα και κάποιες μελλοντικές επεκτάσεις της έρευνας.

2 . Το Διαδίκτυο των Πραγμάτων

2.1 Ιστορική ανασκόπηση

Το Διαδίκτυο των Πραγμάτων (Internet of Things, IoT) κατά τον επιχειρηματία Kevin Ashton, ο οποίος πρώτος απέδωσε τον όρο, αφορά ένα σύστημα αντικειμένων - «πραγμάτων» τα οποία είναι συνδεδεμένα μεταξύ τους, χρησιμοποιώντας το διαδίκτυο ως υπόβαθρο για μια τέτοια σύνδεση, έτσι ώστε να ανταλλάσσουν και να συλλέγουν δεδομένα. (Rose, Eldridge και Chapin (2015)). Το έτος 1997 ο Ashton εξέτασε την δυνατότητα χρήσης ετικετών αναγνώρισης ραδιοσυχνοτήτων (Radio Frequency Identification, RFID), με στόχο την παρακολούθηση της κίνησης προϊόντων, μέσω της εφοδιαστικής αλυσίδας της πολυεθνικής εταιρείας Procter & Gamble. Οι ετικέτες RFID χρησιμοποιήθηκαν για την ανάγνωση και αναγνώριση αντικειμένων και εν συνεχεία για την μετάδοση των πληροφοριών ασύρματα μέσω ενός δικτύου. Νωρίτερα όμως, η “υιοθέτηση” των ετικετών RFID από τις βιομηχανίες, είχε ήδη ξεκινήσει το 1980 (Xu, He, και Li (2014)). Στη συνέχεια, εμφανίστηκε μια νέα ιδέα όπου αισθητήρες και ενεργοποιητές, μέσω των ασυρμάτων δικτύων αισθητήρων (WSN) φάνηκε να ανιχνεύει, να παρακολουθεί και να καταγράφει αντικείμενα, με εφαρμογές στην υγειονομική περίθαλψη και τη διαχείριση της οδικής κυκλοφορίας (Xu, He και Li (2014)). Σήμερα, αυτά τα δίκτυα έχουν εμπλουτιστεί με συσκευές θεσιθεσίας (GPS), «έξυπνων» κινητών τηλεφώνων, με κοινωνικά δίκτυα, με υπολογιστικά νέφη και ανάλυση δεδομένων για την υποστήριξη της σύγχρονης έννοιας του IoT. Στην Εικόνα 2.1-1 φαίνονται κάποιες από τις εφαρμογές του Διαδικτύου των Πραγμάτων όπως: η «έξυπνη» οικία, συνδεδεμένα οχήματα και «έξυπνες» συσκευές.



Εικόνα 2.1-1: Εφαρμογές IoT (IoT Applications, edureka.co)

Στην Ευρώπη, και ιδιαίτερα στη Γερμανία, το IoT είναι μία από τις ιδρυτικές τεχνολογίες της Βιομηχανίας 4.0 στον κατασκευαστικό τομέα. Η Βιομηχανία 4.0 αναφέρεται στην τέταρτη βιομηχανική επανάσταση με τις τρεις πρώτες βιομηχανικές περιστροφές να σχετίζονται με την εκμηχάνιση (Βιομηχανία 1.0), τη μαζική παραγωγή (Βιομηχανία 2.0) και την αυτοματοποίηση (Βιομηχανία 3.0). Οι Zhou, Liu και Zhou (2015) ορίζουν την βιομηχανία 4.0 ως την ενοποίηση των τεχνολογιών πληροφοριακών και επικοινωνιακών συστημάτων με τη βιομηχανική τεχνολογία. Επιπλέον, με την τεχνολογία του IoT, η βιομηχανία 4.0 χρήζει την ανάπτυξη κυβερνοφυσικών συστημάτων (Cyber-Physical System, CPS) και κατασκευής υπολογιστικών νεφών (Cloud Manufacturing, CM). Το CPS συγκροτείται από μηχανήματα, συστήματα αποθήκευσης και εγκαταστάσεις παραγωγής που θα μπορούσαν να ανταλλάσσουν αυτόνομα πληροφορίες, να «πυροδοτούν» ενέργειες και να παρακολουθούνται μεταξύ τους (Kagermann, Wahlster και Helbig (2013)). Σύμφωνα με τους Cheng et al. (2016), ένα CPS συνδέει μια κατασκευαστική οντότητα με εικονικά (υπολογιστικά) και φυσικά (μηχανήματα) στοιχεία, ενσωματώνοντας έτσι αναλογικό και ψηφιακό υλικό. Το IoT παρέχει την απαραίτητη πλατφόρμα για τη σύνδεση του CPS χρησιμοποιώντας ένα δίκτυο αισθητήρων, ενεργοποιητών και συσκευών. Οι πλατφόρμες IoT χρησιμοποιούν γενικά δυνατότητες υπολογιστικού νέφους σε εξωτερικά κέντρα δεδομένων, γεγονός που οδήγησε στην έννοια της κατασκευής cloud (CM) στο πλαίσιο της βιομηχανίας 4.0.

2.2 Ορισμός του IoT

Οι ορισμοί που έχουν δοθεί στη βιβλιογραφία για το Διαδίκτυο των Πραγμάτων γενικά ποικίλουν. Οι πρώτοι ορισμοί του IoT εγκύπτουν περισσότερο στα «πράγματα» (Atzori, Iera και Morabito (2010)) και αφορούν κυρίως ετικέτες RFID που είναι συνδεδεμένες σε ένα δίκτυο για τη μετάδοση πληροφοριών αναγνώρισης (Xu, He και Li (2014)). Αργότερα, σαν «πράγματα» εμφανίστηκαν οι αισθητήρες και ενεργοποιητές οι οποίοι ενσωματώθηκαν στις σημερινές κινητές συσκευές, γενικά. Παρακάτω ακολουθεί μια επιλογή ορισμών που αντικατοπτρίζουν την ποικιλία στην ερμηνεία του IoT:

- I. Ο όρος «Internet-of-Things» χρησιμοποιείται ως λέξη-κλειδί για την κάλυψη διαφόρων πτυχών που σχετίζονται με την επέκταση του Διαδικτύου και του Ιστού στον φυσικό χώρο, μέσω της ευρείας ανάπτυξης χωρικά κατανεμημένων συσκευών με ενσωματωμένη αναγνώριση και δυνατότητες αίσθησης-

ενεργοποίησης. Το Internet-of-Things προβλέπει ένα μέλλον στο οποίο οι ψηφιακές και φυσικές οντότητες μπορούν να συνδεθούν, μέσω κατάλληλων τεχνολογιών πληροφοριών και επικοινωνιών, έτσι ώστε να επιτρέψουν την ανάπτυξη μιας νέας κατηγορίας εφαρμογών και υπηρεσιών (Miorandi et al. (2012)).

- II. Το IoT αφορά μια παγκόσμια δυναμική υποδομή δικτύου με δυνατότητες αυτόματης διαμόρφωσης που βασίζονται σε τυπικά και διαλειτουργικά πρωτόκολλα επικοινωνίας, όπου τα φυσικά και εικονικά «Πράγματα» έχουν ταυτότητες, φυσικά χαρακτηριστικά και εικονικές προσωπικότητες, ενώ χρησιμοποιούν έξυπνες διεπαφές και ενσωματώνονται απρόσκοπτα στο δίκτυο πληροφοριών. (Xu, He και Li (2014)). Το «Διαδίκτυο των Πραγμάτων» είναι μια ομάδα υποδομών που διασυνδέουν συνδεδεμένα αντικείμενα και επιτρέπουν τη διαχείριση, την εξόρυξη δεδομένων και την πρόσβαση στα δεδομένα που δημιουργούν. (Τα συνδεδεμένα αντικείμενα ορίζονται ως αισθητήρες ή/και ενεργοποιητές που εκτελούν μια συγκεκριμένη λειτουργία και μπορούν να επικοινωνούν με άλλο εξοπλισμό. Είναι μέρος μιας υποδομής που επιτρέπει τη μεταφορά, αποθήκευση, επεξεργασία και πρόσβαση στα δεδομένα που παράγονται από χρήστες ή άλλα συστήματα. (Dorsemaine et al. (2015)).
- III. Το «Internet of Things» είναι το δίκτυο των φυσικών αντικειμένων ή «πραγμάτων», ενσωματωμένο με λογισμικά, αισθητήρες, ηλεκτρονικά και συνδεσιμότητα έτσι ώστε να μπορέσει να επιτευχθεί η μεγαλύτερη αξία και η καλύτερη εξυπηρέτηση μέσω ανταλλαγής δεδομένων μεταξύ του κατασκευαστή, του χειριστή ή/και άλλων συνδεδεμένων συσκευών. Κάθε αντικείμενο είναι αναγνωρίσιμο μέσω του ενσωματωμένου συστήματος, αλλά είναι σε θέση να μπορεί να συνεργάζεται με την υφιστάμενη υποδομή του διαδικτύου. (“Internet of Things”, 2020).

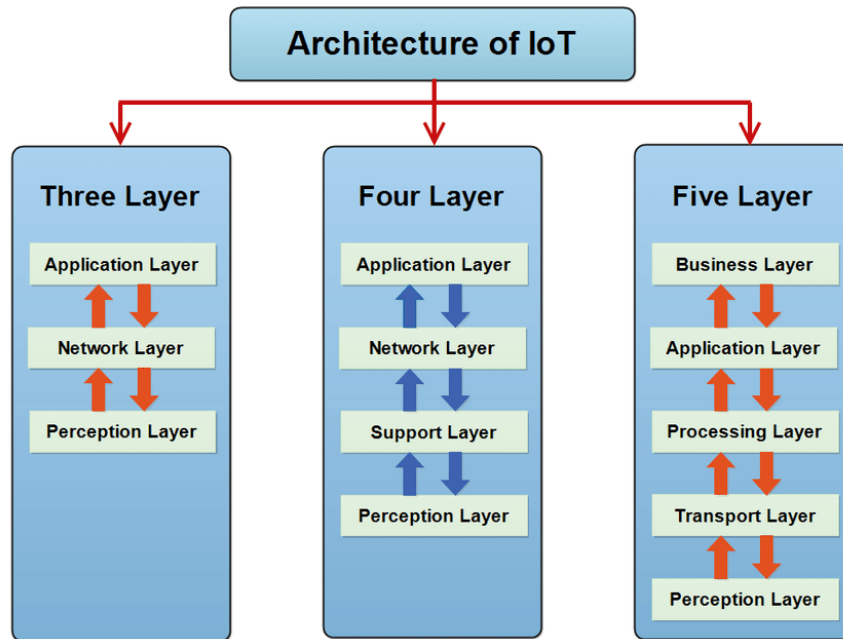
Οι παραπάνω ορισμοί αφορούν σενάρια, στα οποία η συνδεσιμότητα με το δίκτυο και η δυνατότητα των υπολογιστών εκτείνεται σε μια ομάδα αντικειμένων-συσκευών που δεν θεωρούνται υπολογιστικές μονάδες, επιτρέπουν όμως στις συσκευές αυτές να παράγουν και να ανταλλάσσουν δεδομένα με ελάχιστη ανθρώπινη παρέμβαση. Οι διάφοροι ορισμοί που δίνονται για το IoT, τονίζουν τις διαφορετικές πτυχές του, για διαφορετικές περιπτώσεις χρήσης.

2.3 Τεχνολογία του Διαδικτύου των Πραγμάτων

Λόγω της απαίτησης που υπάρχει το IoT να διασυνδέει εκατομμύρια ετερογενή αντικείμενα μέσω του διαδικτύου, γίνεται επιτακτική η ανάγκη για ανάπτυξη μιας ευέλικτης πολυεπίπεδης αρχιτεκτονικής. Ο συνεχής αυξανόμενος αριθμός προτεινόμενων αρχιτεκτονικών δεν έχει καταλήξει σε ένα κοινό μοντέλο αναφοράς. Τα τρία βασικά αρχιτεκτονικά επίπεδα από τα οποία αποτελείται μια εφαρμογή IoT, σύμφωνα με τους Li, Xu και Zhao (2015), κατηγοριοποιούνται όπως παρακάτω:

- Επίπεδο Εφαρμογής, όπου ενσωματώνει διαφορετικούς τύπους «πραγμάτων» όπως ετικέτες RFID, αισθητήρες, ενεργοποιητές.
- Επίπεδο Δικτύωσης, το οποίο υποστηρίζει τη μεταφορά πληροφοριών μέσω ενσύρματου ή ασύρματου δικτύου.
- Επίπεδο Αντίληψης, που εκτελεί την ανάλυση δεδομένων και τη λήψη αποφάσεων μετά τη συλλογή τους

Ωστόσο, στην βιβλιογραφία έχουν προστεθεί και άλλα αρχιτεκτονικά μοντέλα που προσφέρουν καλύτερα αποτελέσματα. Στην Εικόνα 2.3-1 απεικονίζονται κάποιες κοινές αρχιτεκτονικές μεταξύ τους. Υπάρχει το μοντέλο 5 επιπέδων που χρησιμοποιεί μερικώς την αρχιτεκτονική των τριών επιπέδων. Παρακάτω ακολουθεί μια σύντομη αναφορά για αυτά τα πέντε επίπεδα.



Εικόνα 2.3-1: Αρχιτεκτονική IoT (Burhan et al. (2018), The layered architectures of IoT (three, four and five layers))

Επίπεδο Αντίληψης

Το επίπεδο αυτό αφορά τα αντικείμενα (συσκευές που είναι αισθητήρες και ενεργοποιητές) και αποσκοπούν στην συλλογή και επεξεργασία πληροφοριών. Το επίπεδο αντίληψης ψηφιοποιεί και μεταφέρει δεδομένα στο επίπεδο λήψης αντικειμένου μέσω ασφαλών καναλιών. Τα μεγάλα δεδομένα (big data) που παράγονται από το IoT ξεκινούν σε αυτό το επίπεδο (Fuqaha et al. (2015)).

Επίπεδο Λήψης Αντικειμένου

Το επίπεδο λήψης αντικειμένου, προωθεί τα δεδομένα που παράγονται στο επίπεδο αντίληψης, στο επίπεδο διαχείρισης υπηρεσίας μέσω ασφαλών καναλιών. Τα δεδομένα μεταφέρονται χρησιμοποιώντας διάφορες τεχνολογίες όπως RFID, 3G, GSM, UMTS, Wi-Fi, Bluetooth Low Energy, υπέρυθρες, ZigBee κ.α. Επιπλέον κάποιες άλλες λειτουργίες, όπως η υπολογιστική νέφους και διαδικασίες διαχείρισης δεδομένων, εκτελούνται σε αυτό το επίπεδο (Fuqaha et al. (2015)).

Επίπεδο Διαχείρισης Υπηρεσίας

Σε αυτό αυτό το επίπεδο, δίνεται η δυνατότητα στους προγραμματιστές εφαρμογών IoT να δουλεύουν με ετερογενή αντικείμενα, χωρίς να λαμβάνεται υπόψη κάποια συγκεκριμένη πλατφόρμα hardware. Επίσης, αυτό το επίπεδο επεξεργάζεται τα

δεδομένα που εισέρχονται, λαμβάνει αποφάσεις και διανέμει τις απαιτούμενες υπηρεσίες πάνω από πρωτόκολλα καλωδίων δικτύου (Fuqaha et al. (2015)).

Επίπεδο Εφαρμογής

Το επίπεδο εφαρμογής παρέχει τις υπηρεσίες που αιτούνται οι χρήστες. Για παράδειγμα, το επίπεδο εφαρμογής μπορεί να παρέχει μετρήσεις θερμοκρασίας και υγρασίας στον χρήστη που ζητά τα δεδομένα αυτά. Η σημασία αυτού του επιπέδου για το IoT, είναι ότι έχει τη δυνατότητα να παρέχει υψηλής ποιότητας έξυπνες υπηρεσίες για να καλύψει τις ανάγκες των χρηστών. (Fuqaha et al. (2015)).

Επίπεδο Επιχείρησης

Το επίπεδο επιχείρησης διαχειρίζεται τις συνολικές δραστηριότητες και υπηρεσίες του συστήματος IoT. Οι λειτουργίες αυτού του επιπέδου είναι: η δημιουργία ενός επιχειρηματικού μοντέλου, γραφημάτων, διαγραμμάτων ροής κ.λ.π. με βάση τα ληφθέντα δεδομένα από το επίπεδο εφαρμογής. Υποτίθεται επίσης ότι σχεδιάζει, αναλύει, εφαρμόζει, αξιολογεί, παρακολουθεί και αναπτύσσει στοιχεία που σχετίζονται με το σύστημα IoT. Το επίπεδο επιχείρησης καθιστά δυνατή την υποστήριξη διαδικασιών λήψης αποφάσεων με βάση την ανάλυση μεγάλων δεδομένων. Επιπλέον, σε αυτό το επίπεδο επιτυγχάνεται παρακολούθηση και διαχείριση των υποκείμενων τεσσάρων επιπέδων. Ακόμη, αυτό το επίπεδο συγκρίνει την έξοδο κάθε επιπέδου με την αναμενόμενη έξοδο για τη βελτίωση των υπηρεσιών και τη διατήρηση του απορρήτου των χρηστών. (Fuqaha et al. (2015)). Κατά τους Fuqaha et al. (2015), στο μοντέλο πέντε επιπέδων, το επίπεδο εφαρμογής είναι η διεπαφή με την οποία οι τελικοί χρήστες μπορούν να αλληλεπιδρούν με μια συσκευή και να ζητήσουν δεδομένα. Παρέχει επίσης μια διεπαφή στο επίπεδο επιχείρησης, όπου μπορεί να δημιουργηθεί υψηλού επιπέδου ανάλυση και αναφορές. Οι μηχανισμοί ελέγχου για την πρόσβαση σε δεδομένα στο επίπεδο εφαρμογής, διαχειρίζονται επίσης σε αυτό το επίπεδο. Αυτό το επίπεδο φιλοξενείται σε ισχυρές συσκευές λόγω των πολύπλοκων και μεγάλων υπολογιστικών αναγκών του. Λαμβάνοντας υπόψη τα παραπάνω σημεία αφενός και διατηρώντας την απλότητα της αρχιτεκτονικής από την άλλη πλευρά, η αρχιτεκτονική των πέντε επιπέδων είναι το πιο εφαρμόσιμο μοντέλο για εφαρμογές IoT.

2.4 Μοντέλα επικοινωνίας στο Διαδίκτυο των Πραγμάτων

Παρακάτω περιγράφονται τρία διαφορετικά μοντέλα επικοινωνίας που χρησιμοποιούνται στο IoT.

1) Μοντέλο Μηχανή προς Μηχανή (Machine-to-Machine Communication)

Αυτό το μοντέλο επικοινωνίας χρησιμοποιείται σε πολλές συσκευές, οι οποίες μπορούν να συνδέονται και να ανταλλάσσουν πληροφορίες μεταξύ τους απευθείας, χωρίς να είναι απαραίτητη η ενδιάμεση υποστήριξη από hardware (Wortmann και Flüchter (2015)). Αυτές οι συσκευές μπορούν να συνδεθούν μεταξύ τους μέσω διαφορετικών τύπων δικτύων, συμπεριλαμβανομένων του Διαδικτύου ή δικτύων IP. Στην Εικόνα 2.4-1 φαίνεται η επικοινωνία ενός διακόπτη με έναν λαμπτήρα μέσω ασύρματου δικτύου (Bluetooth).

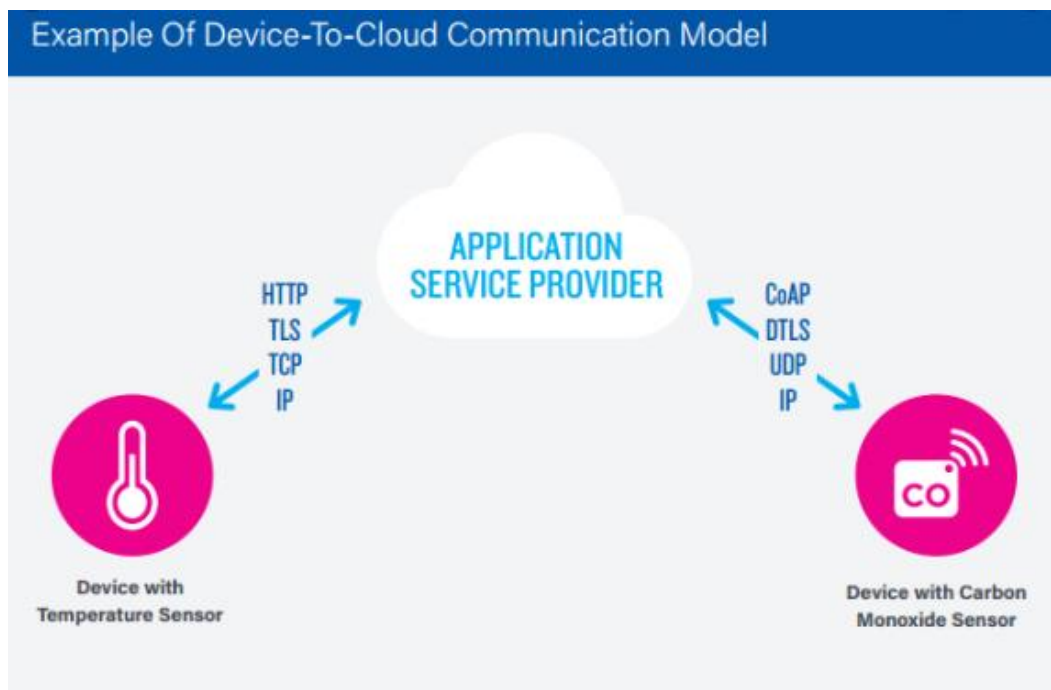


Εικόνα 2.4-1: Μοντέλο επικοινωνίας μηχανή προς μηχανή (<https://www.supinfo.com/articles/single/4235-internet-of-things>, 2017)

Αυτό το μοντέλο χρησιμοποιείται συνήθως σε πολλές εφαρμογές, όπως έξυπνα οικιακά συστήματα ή αυτόματος έλεγχος σε ηλεκτρικά συστήματα, τα οποία επικοινωνούν μεταξύ τους μέσω αποστολής μικρών πακέτων δεδομένων και έχουν σχετικά χαμηλές απαιτήσεις ρυθμού δεδομένων. Οι τυπικές συσκευές IoT αυτού του τύπου είναι έξυπνες κλειδαριές πορτών, έξυπνοι διακόπτες και έξυπνα φώτα, μεταξύ άλλων, τα οποία συνήθως ανταλλάσσουν μικρά πακέτα δεδομένων. Από τη μεριά των χρηστών, το πρόβλημα που εμφανίζεται σε αυτό το μοντέλο είναι η έλλειψη συμβατότητας λόγω των διαφορετικών συσκευών που παράγουν διάφοροι κατασκευαστές χρησιμοποιώντας διαφορετικά πρωτόκολλα. (Wortmann και Flüchter (2015))

2) Μοντέλο Μηχανή προς Νέφος (Machine-to-Cloud Communication)

Σε αυτό το μοντέλο επικοινωνίας μηχανή προς νέφος, οι συσκευές IoT επικοινωνούν και ανταλλάσσουν δεδομένα με υπηρεσίες cloud ή αποθηκεύουν τα δεδομένα σε κάποιον απομακρυσμένο δίσκο, λόγω των περιορισμών της υπολογιστικής ικανότητας και του αποθηκευτικού χώρου των συσκευών. Αυτή η μέθοδος απαιτεί συνήθως βοήθεια από προϋπάρχουσες τεχνολογίες επικοινωνίας, όπως είναι οι συμβατικές ενσύρματες ή ασύρματες συνδέσεις. (Wortmann και Flüchter (2015)). Στην Εικόνα 2.4-2 παρουσιάζεται ένα παράδειγμα του συγκεκριμένου μοντέλου.



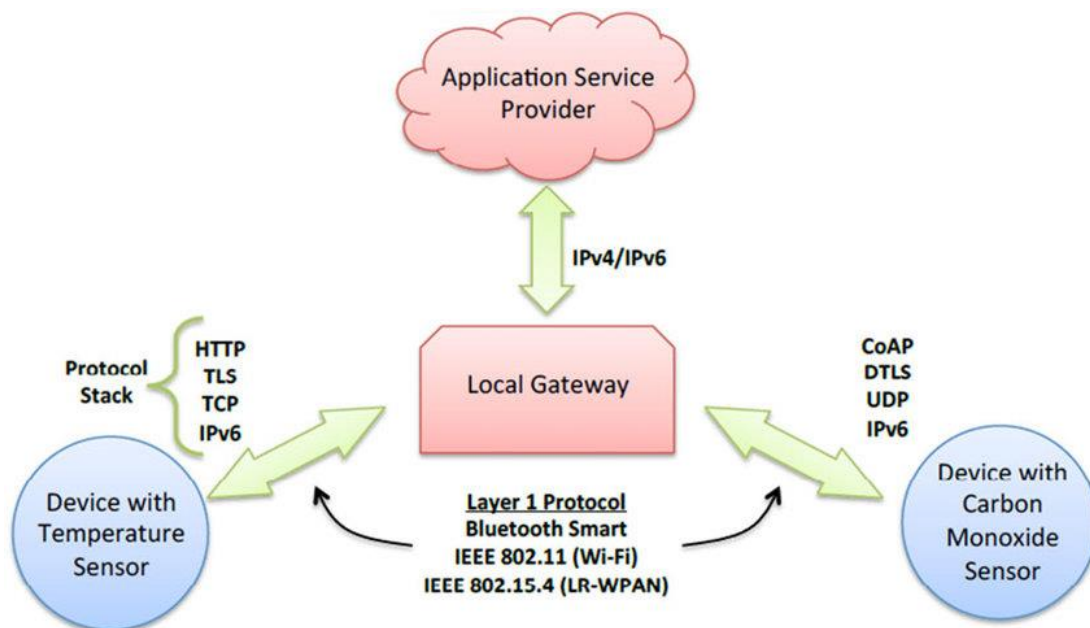
Εικόνα 2.4-2: Μοντέλο επικοινωνίας μηχανή προς νέφος (<https://www.supinfo.com/articles/single/4235-internet-of-things>, 2017)

Αν και το μοντέλο επικοινωνίας μηχανή προς νέφος επιλύει τα προβλήματα του μοντέλου μηχανή προς μηχανή, αυτό το μοντέλο εξαρτάται από το προϋπάρχον δίκτυο. Έτσι κατ' επέκταση, το εύρος ζώνης και οι πόροι του δικτύου περιορίζουν την απόδοση αυτού του μοντέλου. Για τη βελτίωση της απόδοσης του μοντέλου επικοινωνίας μηχανή-νέφος, είναι απαραίτητο να βελτιστοποιηθεί η δομή του δικτύου.

3) Μοντέλο επικοινωνίας Μηχανή προς Πύλη (Machine to Gateway Communication)

Σε αυτό το μοντέλο, μια συσκευή IoT συνδέεται και ανταλλάσει δεδομένα με μια υπηρεσία νέφους, μέσω κάποιας ενδιάμεσης συσκευής που λειτουργεί ως πύλη (Εικόνα

2.4-3). Αυτή η πύλη μπορεί να είναι ένα έξυπνο κινητό τηλέφωνο, στο οποίο έχει εγκατασταθεί μια εφαρμογή και μέσω αυτής επικοινωνεί το τηλέφωνο με την υπηρεσία νέφους και με άλλες συσκευές IoT. Επίσης οι πύλες παρέχουν λειτουργίες όπως ασφάλεια, μείωση της κατανάλωσης ενέργειας των συσκευών IoT (λόγω της μεταφοράς και εκτέλεσης υπολογιστικών εργασιών στην εφαρμογή της ενδιάμεσης πύλης) καθώς και μετάφραση πρωτοκόλλων και δεδομένων. Ακόμη, η χρήση ενδιάμεσης πύλης επιλύει τις προκλήσεις της διεπικοινωνίας μεταξύ των συσκευών διαφορετικών κατασκευαστών (Rose, Eldridge και Chapin (2015)).



Εικόνα 2.4-3: Μοντέλο επικοινωνίας μηχανή προς πύλη (Abdel-Basset et al. (2018), The model of device-to-gateway communication)

2.5 Το Διαδίκτυο των Οχημάτων

Κάθε χρόνο σε παγκόσμιο επίπεδο, προκύπτουν περίπου 8 εκατομμύρια τροχαία ατυχήματα, στα οποία τραυματίζονται 7 εκατομμύρια άνθρωποι και θανατώνονται 1,3 εκατομμύρια άνθρωποι, περίπου. Ακόμη, 90 δισεκατομμύρια ώρες σπαταλούνται λόγω κυκλοφοριακών προβλημάτων (ατυχήματα, κυκλοφοριακή συμφόρηση), η οποία προκαλεί πτώση γύρω στο 2% της συνολικής εγχώριας παραγωγικότητας. Επιπλέον, 220 εκατομμύρια τόνοι ισοδύναμου άνθρακα παράγονται από οχήματα εφοδιασμένα με μηχανές εσωτερικής καύσης (Contreras, Zeadally και Guerrero-Ibanez (2017)). Για τη βελτίωση της απόδοσης του τομέα των μεταφορών, την ενίσχυση της οδικής ασφάλειας

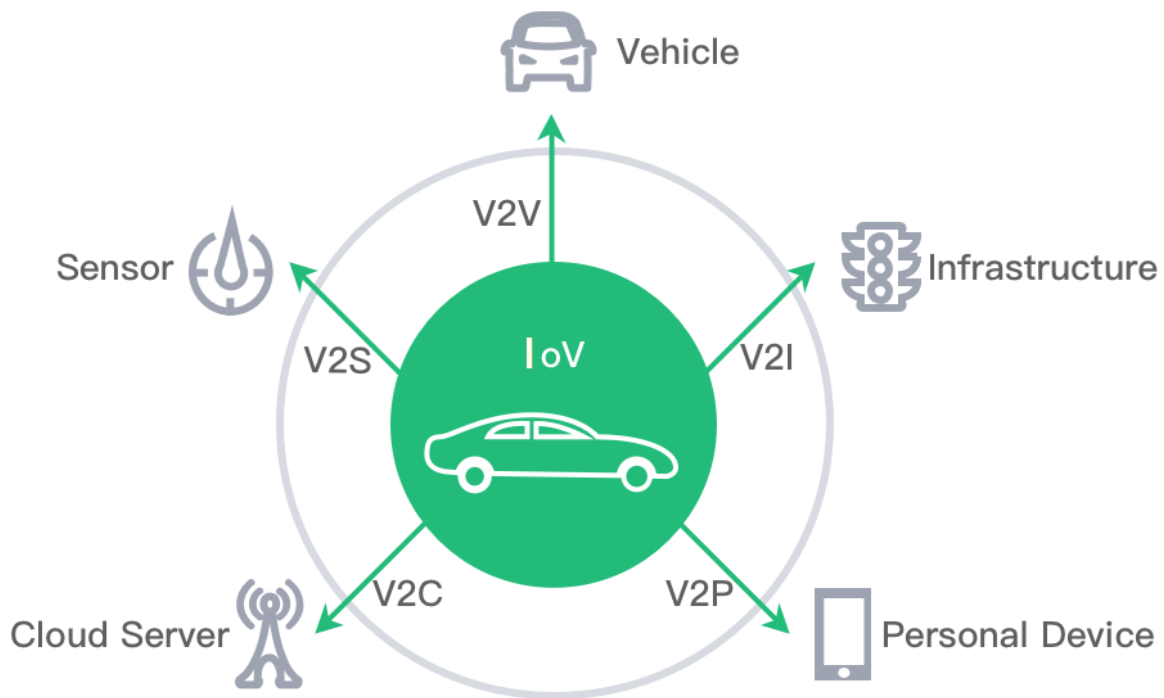
και την προστασία του περιβάλλοντος, εισήχθηκε η έννοια ενός ευφυούς συστήματος μεταφορών (Intelligent Transportation System, ITS). Η εμφάνιση του ITS υποστηρίχθηκε ιδιαίτερα από την πρόοδο των τεχνολογιών αισθητήρων και επικοινωνίας καθώς και από την ανάπτυξη δικτυακών πληροφοριακών συστημάτων, τεχνολογιών λήψης αποφάσεων και φυσικών υποδομών.

Σε όλο τον κόσμο, ο αριθμός των οχημάτων τόσο για ιδιωτική όσο και για εμπορική χρήση ήταν ένα δισεκατομμύριο το 2010 και αναμένεται να είναι 2 δισεκατομμύρια έως το 2030, κατά τον Gross (2016). Η έννοια του Vehicular Ad Hoc Network (VANET) εμφανίστηκε πριν από μια δεκαετία, και έκτοτε υπήρξε ένας πολύ ενεργός τομέας έρευνας (Awang et al. (2017)). Η βασική ιδέα του VANET θεωρεί τα οχήματα ως κινητούς κόμβους που μπορούν να επικοινωνήσουν για τη δημιουργία ενός δικτύου (Contreras, Zeadally και Guerrero-Ibanez (2017)). Βασικά, λόγω περιορισμών κινητικότητας, τα VANET θεωρούνται εξαρτόμενα δίκτυα, όπου η απόδοσή τους επηρεάζεται από την πυκνότητα και την κατανομή των οχημάτων (Darwish και Bakar (2015)), όπως επίσης και από διάφορους άλλους παράγοντες, δηλαδή κακές συμπεριφορές οδηγών και ψηλά κτίρια (Contreras, Zeadally και Guerrero-Ibanez (2017)). Επιπλέον, τα οχήματα θεωρούνται ασταθής, προσωρινοί και τυχαίοι κόμβοι. Έτσι, ένα VANET δεν μπορεί να εγγυηθεί βιωσιμότητα εφαρμογών / υπηρεσιών για πελάτες σε περιοχές μεγάλης κλίμακας (Fangchun et al. (2014)). Επομένως, το VANET είναι πιο κατάλληλο για εφαρμογές περιορισμένης κλίμακας που απαιτούν υπηρεσίες ad hoc όπως, αποτροπή συγκρούσεων ή ειδοποίηση του οδηγού για πιθανό κίνδυνο στους δρόμους. Ωστόσο, με την ανάπτυξη της τεχνολογίας του Διαδικτύου των Πραγμάτων (IoT) και την αύξηση του αριθμού των συνδεδεμένων οχημάτων στο Διαδίκτυο, απαιτούνται νέα κριτήρια για τα VANET. Ένα ακόμη μειονέκτημα των VANET είναι οι περιορισμένες δυνατότητές που έχουν να επεξεργάζονται όλες τις πληροφορίες που συλλέγονται από τα ίδια και τους γύρω παράγοντες (όπως είναι κινητές συσκευές και αισθητήρες). Για να εξυπηρετήσουν τις νέες απαιτήσεις των ITS, τα οχήματα πρέπει να λειτουργούν ως μια έξυπνη πλατφόρμα πολλαπλών αισθητήρων με: συνδεσιμότητα στο Διαδίκτυο που βασίζεται σε πρωτόκολλο διαδικτύου (Internet Protocol, IP), ισχυρά υπολογιστικά στοιχεία και ικανότητα επικοινωνίας με άλλα οχήματα και συσκευές ITS (Lee et al. (2016)). Κατά τους Kaiwartya et al. (2016) η εξέλιξη της έννοιας των VANET οδήγησε στην εισαγωγή μιας νέας έννοιας του Διαδικτύου των Οχημάτων (Internet of Vehicles, IoV). Έτσι, ως ξεχωριστή περίπτωση του IoT, το IoV έχει ξεχωριστά

χαρακτηριστικά και ειδικές απαιτήσεις για την εξυπηρέτηση των ευφυών συστημάτων μεταφοράς. Σύμφωνα με τους Nitti et al. (2014), το IoV ορίζεται ως μια πλατφόρμα που πραγματοποιεί σε βάθος την ενοποίηση και ανταλλαγή πληροφοριών μεταξύ ανθρώπων, οχημάτων, πραγμάτων και περιβάλλοντος. Ο κύριος στόχος του IoV είναι να ενισχύσει την ασφάλεια και την αποτελεσματικότητα των μεταφορών, να βελτιώσει το επίπεδο υπηρεσιών των πόλεων, να προφυλάξει το περιβάλλον, και να διασφαλίσει ότι οι άνθρωποι είναι ικανοποιημένοι με τις υπηρεσίες μεταφορών. Σε αντίθεση με τα VANET, το IoV ενσωματώνει την ευφυΐα οχημάτων με την δικτύωση στα οχήματα, κάτι που έχει ως αποτέλεσμα την εμφάνιση ευφυών δικτύων με επικοινωνιακές και υπολογιστικές δυνατότητες που παρέχουν υπηρεσίες μεταφορών σε περιοχές μεγάλης κλίμακας (Fangchun et.al. (2014)). Στο περιβάλλον του IoV, καθώς τα οχήματα έχουν σταθερή σύνδεση στο Διαδίκτυο, μπορούν να παρέχουν πληροφορίες για διάφορες εφαρμογές των ITS όπως οδική ασφάλεια, προληπτική συντήρηση του οχήματος και καλή λειτουργική κατάσταση, διαχείριση της κυκλοφορίας και ψυχαγωγία. Κατά συνέπεια, η ανταλλαγή δεδομένων ενεργοποιείται μεταξύ αισθητήρων και ενεργοποιητών, οδικών υποδομών και οχημάτων καθώς και οδηγών και επιβατών. Έτσι το IoV συλλέγει μεγάλο όγκο δεδομένων διαφορετικού περιεχομένου από μια περιοχή μεγάλης κλίμακας, το οποίο έχει ως επακόλουθο την έννοια των μεγάλων δεδομένων. Με τα σημαντικά πλεονεκτήματα που διαθέτει το IoV έναντι των VANET ανοίγονται πολλές νέες ευκαιρίες. Μέσω της ανάπτυξης του IoV, οι πάροχοι υπηρεσιών μπορούν να εισάγουν νέες υπηρεσίες μεταφορών, όπως παρακολούθηση της κυκλοφορίας σε πραγματικό χρόνο, παρακολούθηση της κατάστασης του οχήματος όσον αφορά την συντήρηση, εντοπισμός παραβάσεων του κώδικα οδικής κυκλοφορίας, αναφορά συνθηκών σύγκρουσης οχημάτων (πραγματογνωμοσύνη), εντοπισμός χώρων στάθμευσης και εξυπηρέτηση πελατών βάσει τοποθεσίας. Συνεπώς, το IoV προσφέρει διάφορα οφέλη για τους οδηγούς, τις κοινωνίες και τις οικονομίες.

Το IoV αποτελεί αναπόσπαστο τμήμα του IoT, το οποίο συνδέει διάφορα ετερογενή δίκτυα, για παράδειγμα, δίκτυα μεταξύ διαφορετικών οχημάτων και εσωτερικού δικτύου ενός οχήματος, με το Διαδίκτυο. Όπως παρουσιάζεται στην Εικόνα 2.5-1 αυτή η ετερογενής αρχιτεκτονική του δικτύου μπορεί να κατηγοριοποιηθεί σε πέντε τύπους επικοινωνίας οχήματος: όχημα σε όχημα (Vehicle to Vehicle, V2V), όχημα προς υποδομή (Vehicle to Infrastructure, V2I), όχημα προς νέφος (Vehicle to Cloud,

V2C), όχημα προς αισθητήρες (Vehicle to Sensors, V2S) και όχημα προς πεζούς (Vehicle to Pedestrian, V2P).



Εικόνα 2.5-1: Τύποι επικοινωνίας του οχήματος στο IoV (<https://coleofduty.com/market-reports/2020/06/05/global-internet-of-vehicles-market-2020-impact-of-covid-19-future-growth-analysis-and-challenges>)

Σκοπός του IoV είναι να διασφαλίσει ότι οι συσκευές μπορούν να επικοινωνούν μεταξύ τους και να ανταλλάσσουν πληροφορίες με νοημοσύνη, έτσι ώστε να εξυπηρετούν στα προβλήματα που αναφέρθηκαν στην προηγούμενη παράγραφο. Επιπλέον, η έρευνα και ανάπτυξη νέων τεχνολογιών αυτής της υποδομής ενσωματώνει όλο και περισσότερο τεχνολογίες των πληροφοριακών συστημάτων στα οχήματα όπως για παράδειγμα η υλοποίηση της αυτόνομης μοτοσυκλέτας από την BMW η οποία κινείται χωρίς αναβάτη!

3 Υπολογιστικό νέφος (Cloud Computing)

3.1 Τι είναι το υπολογιστικό νέφος;

Η σημερινή τεχνολογία που ονομάζεται υπολογιστικό νέφος είναι η συνέχεια της παλαιότερης τεχνολογίας Πάροχος Υπηρεσιών Εφαρμογών (Application Service Provider, ASP). Στην ουσία πρόκειται για μια εφαρμογή παροχής υπηρεσιών. Αυτή η

εφαρμογή παρέχει υπηρεσίες μέσω δικτύου σε πελάτες, όπως η πρόσβαση σε μια εφαρμογή λογισμικού, κάνοντας χρήση ενός τυποποιημένου πρωτοκόλλου όπως είναι το Hyper Text Transfer Protocol (HTTP). Το νέφος (Cloud) αποτελείται από τεχνολογίες οι οποίες στο ASP δεν υφίστανται. Το Cloud είναι ένα μοντέλο που επιτρέπει, ανεξαρτήτως τοποθεσίας, την πρόσβαση στο δίκτυο, ενός αριθμού διαμορφωμένων υπολογιστικών πόρων που μπορούν να παρέχονται γρήγορα και με ελάχιστη προσπάθεια διαχείρισης ή αλληλεπίδρασης με τον πάροχο υπηρεσιών (Mell και Grance (2011)). Παραδείγματα πόρων αποτελούν:

- I. Αποθήκευση
- II. Επεξεργασία
- III. Μνήμη
- IV. Το εύρος ζώνης του δικτύου (Bandwidth)
- V. Εικονικές μηχανές

3.2 Υπηρεσίες και μοντέλα ανάπτυξης Υπολογιστικού Νέφους

Οι πάροχοι υπολογιστικού νέφους προσφέρουν τις παρακάτω υπηρεσίες (Εικόνα 3.2-1) οι οποίες προσφέρουν διαφορετικές δυνατότητες.

Λογισμικό ως υπηρεσία (Software as a Service, SaaS)

Αφορά την δυνατότητα που παρέχεται στον χρήστη να χρησιμοποιεί τις εφαρμογές που παρέχονται από μια υποδομή Cloud. Οι εφαρμογές αυτές είναι διαθέσιμες μέσω κάποιων διεπαφών διαδικτύου, όπως ένας περιηγητής ιστού (web browser) σε διάφορες υπολογιστικές συσκευές (π.χ. προσωπικούς υπολογιστές, tablets, smartphones και άλλες φορητές συσκευές). Ο χρήστης δεν μπορεί να επηρεάσει τους εξυπηρετητές, τα λειτουργικά συστήματα ή τους αποθηκευτικούς χώρους τους και ως επί το πλείστον δεν έχει καθόλου έλεγχο επί της εφαρμογής. Το Gmail, το Dropbox, το Yahoo Mail είναι κάποια παραδείγματα ευρέως γνωστών εφαρμογών SaaS.

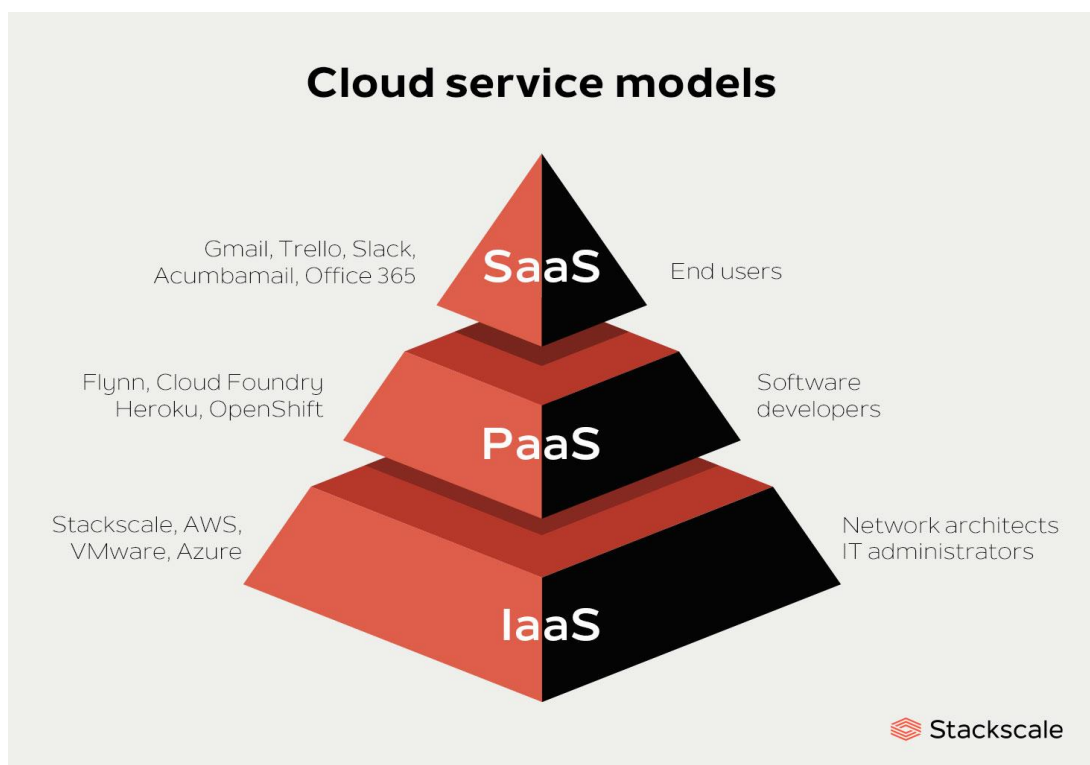
Πλατφόρμα ως υπηρεσία (Platform as a Service, PaaS)

Σε αυτή την υπηρεσία δίνεται η δυνατότητα στον χρήστη να αναπτύσσει πάνω στο Cloud, εφαρμογές που έχει είτε δημιουργήσει είτε αποκτήσει, οι οποίες αναπτύχθηκαν με χρήση γλωσσών προγραμματισμού και εργαλείων που υποστηρίζονται από την πάροχο εταιρεία. Όπως στην υπηρεσία SaaS, έτσι και εδώ ο χρήστης δεν διαχειρίζεται ούτε ελέγχει το σχετικό Cloud που περιέχει τα δίκτυα, τους διακομιστές, τα

λειτουργικά συστήματα ή τα αποθηκευτικά μέσα, αλλά έχει τον έλεγχο των εφαρμογών που έχουν δημιουργηθεί και ίσως των παραμετροποιήσεων του περιβάλλοντος που εμπεριέχει τις εφαρμογές.

Υποδομή ως υπηρεσία (Infrastructure as a Service, IaaS)

Στην IaaS ο χρήστης έχει την δυνατότητα να δεσμεύει προς χρήση επεξεργαστική ισχύ, αποθηκευτικά μέσα, δίκτυα, και άλλους υπολογιστικούς πόρους, καθώς επίσης είναι σε θέση να αναπτύξει και να εκτελέσει αυθαίρετο λογισμικό, το οποίο μπορεί να περιλαμβάνει λειτουργικά συστήματα και εφαρμογές. Ο χρήστης δεν διαχειρίζεται, ούτε ελέγχει το Cloud, αλλά ελέγχει τα λειτουργικά συστήματα, τα αποθηκευτικά μέσα, τις εφαρμογές που έχουν αναπτυχθεί και ενδεχομένως κάποιον περιορισμένο έλεγχο συγκεκριμένου εξοπλισμού δικτύωσης (Ameen και Rahman (2015)).



Εικόνα 3.2-1: Μοντέλα υπηρεσιών υπολογιστικού νέφους, (<https://www.stackscale.com/blog/cloud-service-models/>)

Επιπρόσθετα των μοντέλων παροχής υπηρεσιών νέφους (SaaS, PaaS και IaaS), υφίστανται και τέσσερα μοντέλα ανάπτυξης υπηρεσιών. Τα μοντέλα αυτά είναι τα εξής:

Ιδιωτικό νέφος (Private cloud):

Η δομή αυτού του μοντέλου λειτουργεί αποκλειστικά για έναν οργανισμό. Διαχειριστής μπορεί να είναι ο ίδιος ο οργανισμός ή κάποιο τρίτο μέρος ενώ η υποδομή

μπορεί να βρίσκεται εκτός ή εντός των εγκαταστάσεων του οργανισμού. Όπως είναι αναμενόμενο, τα ιδιωτικά νέφη αποδίδουν καλύτερα σε θέματα ασφάλειας των δεδομένων, λόγω του ότι είναι ιδιωτικά (Mell και Grance (2011)).

Δημόσιο νέφος (Public cloud)

Σε αυτή την περίπτωση η υποδομή του νέφους προσφέρεται στο ευρύ κοινό από κάποιον οργανισμό που διαθέτει τις υπηρεσίες έναντι κόστους. Οι πιο κοινές μορφές του δημοσίου νέφους προσφέρονται μέσω Διαδικτύου (Mell και Grance (2011)).

Κοινοτικό νέφος (Community cloud)

Αυτή η υποδομή μοιράζεται μεταξύ διαφόρων οργανισμών και υποστηρίζει μια συγκεκριμένη κοινότητα που έχει κοινές ανησυχίες (π.χ. αποστολή, απαιτήσεις ασφαλείας, πολιτική και θέματα συμμόρφωσης). Η διαχείρισή της κοινότητας μπορεί να γίνεται από τον ίδιο τον οργανισμό ή από τρίτους και μπορεί να βρίσκεται εντός ή εκτός των εγκαταστάσεων του οργανισμού (Mell και Grance (2011)).

Υβριδικό νέφος (Hybrid cloud)

Η υποδομή υβριδικού νέφους αφορά την σύνθεση δύο ή περισσότερων cloud (δημοσίου, ιδιωτικού ή κοινοτικού) που παραμένουν ως μοναδικές οντότητες. Διατηρούν δηλαδή τα στοιχεία της ανεξαρτησίας τους, αλλά παράλληλα συνδέονται μεταξύ τους με τυποποιημένη ή αποκλειστική τεχνολογία, η οποία επιτρέπει τη φορητότητα δεδομένων και εφαρμογών (Mell και Grance (2011)).

3.3 Οφέλη χρήσης υπολογιστικού νέφους

Η τεχνολογία της υπολογιστικής νέφους υπάρχει εδώ και περίπου δύο δεκαετίες και παρά τα δεδομένα που δείχνουν την επιχειρηματική αποδοτικότητα, τα οφέλη σχετικά με τα κόστη και τα ανταγωνιστικά πλεονεκτήματα που διαθέτει, ένα μεγάλο μέρος της επιχειρηματικής κοινότητας συνεχίζει να λειτουργεί χωρίς την χρήση της. Σύμφωνα με μια μελέτη της International Data Group, το 69% των επιχειρήσεων χρησιμοποιούν ήδη τεχνολογίες Cloud, ενώ το 18% δηλώνουν ότι σχεδιάζουν να εφαρμόσουν λύσεις υπολογιστικού νέφους. Όπως φαίνεται, ένας αυξανόμενος αριθμός τεχνολογικών επιχειρήσεων αναγνωρίζουν τα οφέλη της χρήσης υπολογιστικού νέφους. Οι περισσότεροι οργανισμοί χρησιμοποιούν αυτή την τεχνολογία για να λειτουργούν αποτελεσματικότερα, να προσφέρουν καλύτερη πελατειακή εξυπηρέτηση και να αυξάνουν τα συνολικά περιθώρια κέρδους τους.

Τα βασικά οφέλη της χρήσης των Clouds είναι: η ασφάλεια, η παράλληλη επεξεργασία, οι εικονικοποιημένοι πόροι και η ενσωμάτωση υπηρεσιών ανάλυσης δεδομένων με κλιμακωτή αποθήκευση δεδομένων. Έτσι, η υπολογιστική νέφους μειώνει το κόστος και τους περιορισμούς που έχουν οι εταιρείες για την μηχανογράφηση και την αυτοματοποίηση, αλλά επίσης μειώνει το κόστος συντήρησης της υποδομής και παρέχει πρόσβαση στους χρήστες από παντού με αποτελεσματική διαχείριση των προσφερομένων υπηρεσιών. (Lu et al. (2013)). Συνεπώς, η τεχνολογία νέφους, με το πλούσιο σύνολο-αποθηκευτικού χώρου, υποδομών και υπολογιστικών υπηρεσιών-παρέχει ένα πολύ ενδιαφέρον περιβάλλον για επιστημονικά πειράματα, ανάλυση δεδομένων και ερευνητική δραστηριότητα (Gunarathne et al. (2013)).

3.4 Ασφάλεια στο υπολογιστικό νέφος

Οι πάροχοι υπηρεσιών νέφους καθώς και οι χρήστες, έρχονται αντιμέτωποι με προβλήματα εμπιστευτικότητας, ακεραιότητας δεδομένων, διαθεσιμότητας και ανθεκτικότητας. Κατά συνέπεια τίθενται ζητήματα ασφάλειας στην υποδομή, στον έλεγχο πρόσβασης, στη διαχείριση ταυτότητας και στον έλεγχο ακεραιότητας. Επειδή τα Cloud είναι προσβάσιμα από απόσταση προκύπτουν κάποια θέματα προς αντιμετώπιση όπως π.χ. η υποκλοπή και η άρνηση υπηρεσιών (Denial of Service), καθιστώντας τα επίπεδα δικτύου, εφαρμογής και υποδομής πιο ευάλωτα (Duta (2016)). Οι απειλές της ασφάλειας ενδέχεται να προκύψουν σε οποιοδήποτε στάδιο επεξεργασίας των δεδομένων, συμπεριλαμβανομένων των: συλλογής δεδομένων, φιλτραρίσματος πληροφοριών, μοντελοποίησης και ενσωμάτωσης δεδομένων (data integration). Επιπλέον, η εκθετική αύξηση του ρυθμού παραγωγής δεδομένων δεν μπορεί να εγγυηθεί ασφάλεια σε κρίσιμες πληροφορίες. Ακόμη, οι διαθέσιμες λύσεις σε ζητήματα ασφαλείας έχουν σχεδιαστεί για στατικά σύνολα δεδομένων και όχι για δυναμική και συνεχή παραγωγή δεδομένων (Marjani et. al. (2017)). Επομένως, όσον αφορά το επίπεδο ασφαλείας στο Cloud, εφόσον παρέχει υπολογιστικές υπηρεσίες και αποθηκευτικό χώρο, η ασφάλεια και η αξιοπιστία πρέπει να βελτιστοποιούνται διαρκώς.

3.5 Το υπολογιστικό νέφος στα συνδεδεμένα οχήματα

Το υπολογιστικό νέφος στα οχήματα περιέχει όλα τα απαιτούμενα πρωτόκολλα και τις υπηρεσίες ώστε το όχημα να λειτουργεί αποτελεσματικά και με ασφάλεια. Το Cloud παρέχει ένα υπολογιστικό και επικοινωνιακό περιβάλλον, έτσι ώστε να διασυνδέονται όλα τα Αντικείμενα που φέρει το όχημα και έχουν αίσθηση και κίνηση

(αισθητήρες-ενεργοποιητές). Ένα από τα σημαντικότερα οφέλη του Cloud των οχημάτων θα είναι η αυτόνομη οδήγηση.

Τα αυτόνομα οχήματα θα πρέπει να είναι ικανά να αισθάνονται το περιβάλλον γύρω από αυτά κατά την κίνηση τους, παρουσία αλλαγών λόγω πεζών και άλλων εμποδίων, στοχεύοντας σε μια αποτελεσματική και ασφαλή διαδρομή. Επιπλέον, αυτά τα συστήματα πρέπει να είναι σε θέση να αντιμετωπίζουν απρόσμενα γεγονότα όπως οδικά έργα, κυκλοφοριακή συμφόρηση και ατυχήματα. Έτσι, τα αυτόνομα οχήματα χρειάζονται λεπτομερείς πληροφορίες σχετικά με το περιβάλλον γύρω τους (Maurelli et al. (2009)). Συνήθως, χρησιμοποιούν ενσωματωμένους αισθητήρες όπως radar, lidar, gps, laser για να δημιουργήσουν ένα τρισδιάστατο σημείο-σύννεφο που αντιπροσωπεύει τις συντεταγμένες σημείων (x, y, z) της επιφάνειας των εμποδίων (Εικόνα 3.5-1). Ωστόσο, αυτοί οι αισθητήρες δεν μπορούν να παρέχουν στα οχήματα πληροφορίες σχετικά με κρυμμένα εμπόδια που δεν είναι άμεσα ανιχνεύσιμα, όπως κάποιο σταθμευμένο όχημα πάνω σε στροφή. Επιπλέον, αυτοί οι αισθητήρες έχουν περιορισμένο εύρος και ως εκ τούτου δεν μπορούν να καταγράψουν μεγάλου εύρους δεδομένα με επαρκή ακρίβεια, περιορίζοντας έτσι την ικανότητα του οχήματος.



Εικόνα 3.5-1: Χαρτογράφηση οδικού περιβάλλοντος από το αυτόνομο όχημα (<https://www.smallworldsocial.com/how-autonomous-cars-map-the-environment>)

Από τα δύο παραπάνω ζητήματα, προκύπτει η ανάγκη, τα αυτόνομα οχήματα να έχουν πρόσβαση το ένα στις πληροφορίες του άλλου ώστε να ενισχυθεί ακρίβεια και η ασφάλεια στην οδήγηση (Campbell et al. (2010)).

Μια ομάδα του MIT, οι Swarun, Gollakota και Katabi (2012) έχει αναπτύξει ένα σύστημα νέφους-υποβοήθησης, το Carcel, το οποίο έχει πρόσβαση στα δεδομένα των

αισθητήρων των αυτόνομων οχημάτων και σε στατικούς αισθητήρες στην υποδομή του οδικού δικτύου. Επιπλέον, στο Cloud καταγράφεται η τροχιά που ακολουθεί το κάθε όχημα, συγκεντρώνονται οι πληροφορίες για κάθε διαδρομή και στη συνέχεια μεταφέρει σε κάθε όχημα πληροφορίες σχετικές με πιθανά εμπόδια, τυφλά σημεία και εναλλακτικές διαδρομές. Έτσι, το Cloud έχει την δυνατότητα να βοηθάει τα οχήματα να λαμβάνουν έξυπνες αποφάσεις προβλέποντας εμπόδια που δεν είναι άμεσα ανιχνεύσιμα από τους αισθητήρες του οχήματος. Περαιτέρω το Cloud, έχοντας μεγαλύτερο εύρος πληροφοριών σχετικά με τα μοτίβα κυκλοφορίας και άλλα απρόσμενα γεγονότα όπως ατυχήματα, το Carcel επιτρέπει στα αυτόνομα οχήματα τον σχεδιασμό αποτελεσματικότερων διαδρομών. Για να υλοποιηθεί το σύστημα Carcel, θα πρέπει να προσδιοριστούν δυο προκλήσεις, α) της μεταφοράς δεδομένων αισθητήρων δυναμικά μέσω ασύρματης σύνδεσης περιορισμένου εύρους ζώνης και β) μείωση της καθυστέρησης (latency) και της απώλειας πακέτων.

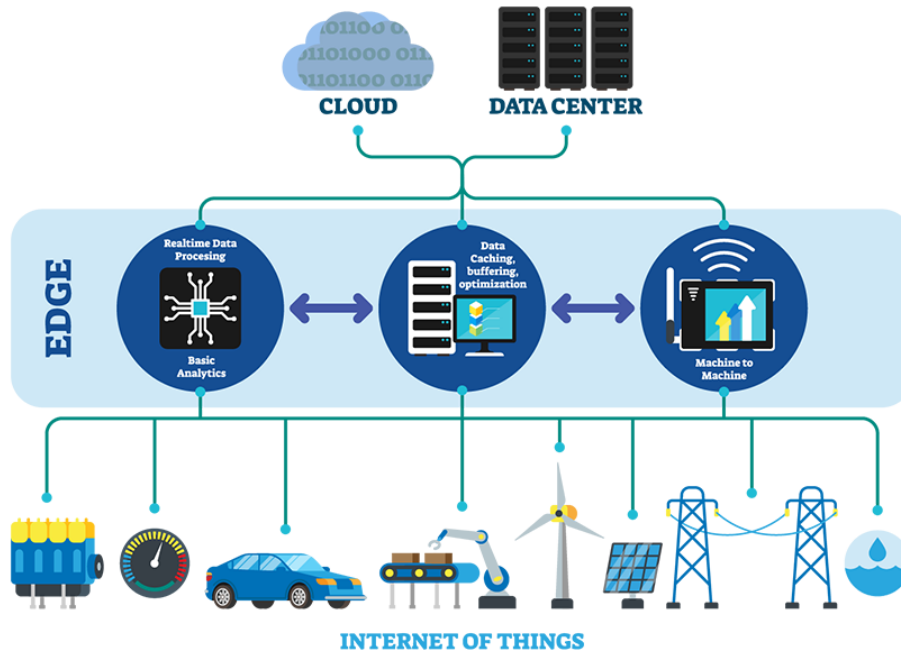
Συμπερασματικά, η χρήση του Cloud σε συνεργασία με το IoV στον τομέα των μεταφορών και των οχημάτων μπορεί να προσφέρει περαιτέρω ασφάλεια κατά την οδήγηση, βελτίωση των οδικών συνθηκών, μείωση της κατανάλωσης καυσίμου ή ενέργειας σε περίπτωση ηλεκτρικών οχημάτων, μείωση εκπεμπόμενων ρύπων και λεπτομερέστερη παρακολούθηση της κατάστασης του οχήματος (προληπτική συντήρηση, πιθανές βλάβες). Το Cloud θα είναι σε θέση να παρέχει το περιβάλλον ενός συστήματος, για την συλλογή δεδομένων από μια «συνεργασία» διαφόρων αισθητήρων καθώς και για εφαρμογές διαμοιρασμού βάσεων δεδομένων, στα συνδεδεμένα οχήματα.

3.6 Υπολογιστική Ομίχλης (Fog Computing)

Η Υπολογιστική Ομίχλης ή πιο γνωστή ως Υπολογιστική Άκρου (Edge Computing) προτείνει, ότι η ευφυΐα και η λειτουργικότητα ενός συστήματος μπορούν να εφαρμοστούν κοντά στην πηγή των δεδομένων, σε αντίθεση με την προσέγγιση της υπολογιστικής νέφους. Καθώς οι όροι «ομίχλη» και «νέφος» χρησιμοποιούνται συνήθως εναλλάξ, παρατίθενται οι έννοιες ως εξής: Η έννοια της Υπολογιστικής Άκρου (Edge Computing) φέρνει την επεξεργασία εγγύτερα στην πηγή δεδομένων και δεν χρειάζεται να σταλεί σε κάποιο απομακρυσμένο νέφος ή σε άλλα κεντρικά συστήματα για επεξεργασία. Η ιδέα της Υπολογιστικής Ομίχλης εισήχθη, από την Cisco το 2014 ως ορολογία, για να ορίσει την Υπολογιστική Άκρου ως πρότυπο, το οποίο καθορίζει τον τρόπο με τον οποίο πρέπει να λειτουργεί η Υπολογιστική Άκρου και παράλληλα

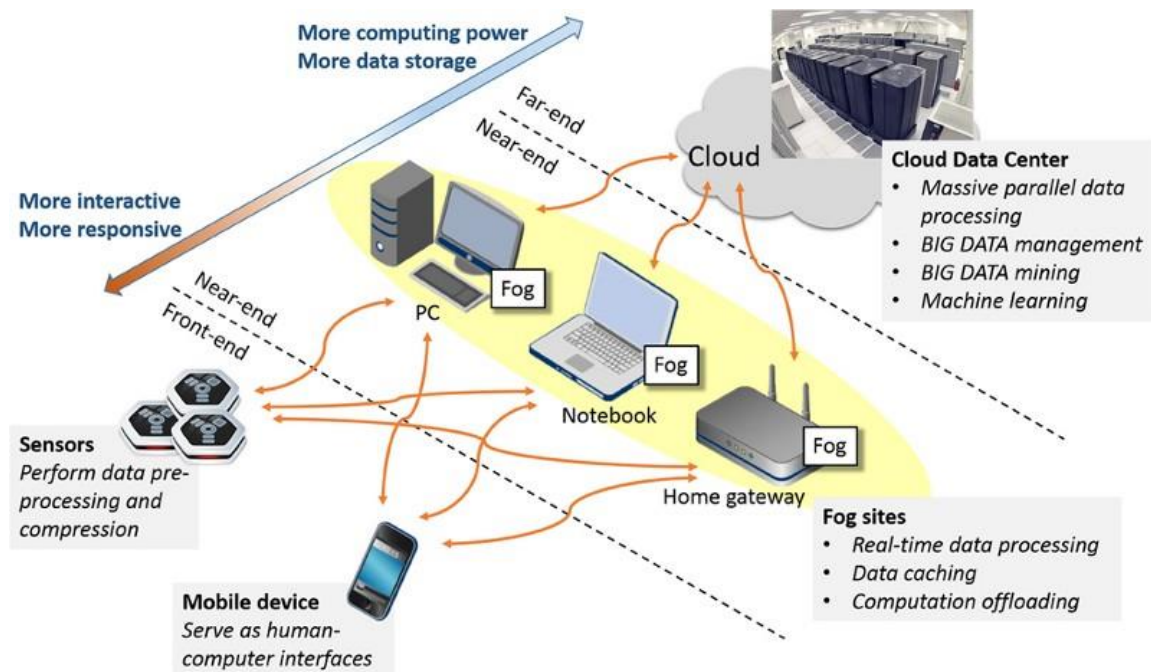
διευκολύνει τη λειτουργία υπηρεσιών υπολογισμού, αποθήκευσης και δικτύωσης μεταξύ τελικών συσκευών και κέντρων δεδομένων Υπολογιστικής Νέφους (David Linthicum (2019)).

Edge Computing



Εικόνα 3.6-1: Χρήση Edge Computing (<https://innovationnetwork.ieee.org/real-life-edge-computing-use-cases>)

Στην Υπολογιστική Άκρου, αντί να μεταφέρεται ο μεγάλος όγκος δεδομένων που παράγεται από διάφορες συσκευές IoT στο Cloud, τα δεδομένα αυτά επεξεργάζονται στην Άκρη του δικτύου (Εικόνα 3.6-1) και έτσι εξοικονομείται εύρος ζώνης δικτύου (bandwidth) και μειώνονται οι καθυστερήσεις στην επικοινωνία (Vallati et al. (2016)). Οι διακομιστές της Υπολογιστικής Άκρου βρίσκονται πιο κοντά στον χρήστη σε αντίθεση με τους διακομιστές νέφους. Έτσι, αν και οι διακομιστές Άκρου έχουν μικρότερη υπολογιστική δύναμη από αυτούς των Cloud, συνεχίζουν να παρέχουν καλύτερη ποιότητα υπηρεσιών (Quality of Service, QoS) και μικρότερη καθυστέρηση στους τελικούς χρήστες. Στην Εικόνα 3.6-2 παρουσιάζεται η δομή των τριών πτυχών του Υπολογιστικού Άκρου front-end, near-end και far-end, οι οποίες περιγράφονται παρακάτω.



Εικόνα 3.6-2: Δομή Edge Computing (Front, Near, Far-End),
 (<https://www.energomonitor.com/insight/edge-computing-fog-computing-benefits-of-fogging/>)

- **FRONT-END**

Σε αυτό το επίπεδο οι τελικές συσκευές (αισθητήρες, ενεργοποιητές) αναπτύσσονται στο front-end της δομής Edge. Το περιβάλλον front-end μπορεί να προσφέρει καλύτερη διαδραστικότητα και ανταπόκριση για τους τελικούς χρήστες. Με την υπολογιστική ισχύ που παρέχεται από την πληθώρα των κοντινών τελικών συσκευών, το Edge μπορεί να παρέχει υπηρεσίες σε πραγματικό χρόνο για ορισμένες εφαρμογές. Ωστόσο, λόγω της περιορισμένης χωρητικότητας των συσκευών, δεν μπορούν να ικανοποιηθούν πολλές απαιτήσεις στο περιβάλλον frontend.

- **NEAR-END**

Οι πύλες δικτύωσης (gateways) που αναπτύσσονται στο περιβάλλον front-end μπορούν να υποστηρίξουν το μεγαλύτερο μέρος της κίνησης στα δίκτυα. Οι διακομιστές Edge μπορούν επίσης να έχουν πολλές απαιτήσεις πόρων, όπως είναι η επεξεργασία δεδομένων σε πραγματικό χρόνο και η προσωρινή αποθήκευση δεδομένων. Στο Edge ένα μεγάλο μέρος αποθήκευσης και επεξεργασίας δεδομένων μεταφέρεται στο περιβάλλον near-end με αποτέλεσμα οι τελικοί χρήστες να πετυχαίνουν καλύτερη απόδοση στις προηγούμενες λειτουργίες, με μια μικρή αύξηση της καθυστέρησης.

- **FAR-END**

Καθώς οι διακομιστές Cloud αναπτύσσονται πιο μακριά από το τελικές συσκευές, η καθυστέρηση μετάδοσης στο δίκτυο είναι σημαντική. Παρ' όλα αυτά, οι διακομιστές Cloud στο περιβάλλον far-end μπορούν να παρέχουν μεγαλύτερη υπολογιστική ισχύ και αποθηκευτικούς χώρους. Για παράδειγμα, οι διακομιστές Cloud είναι σε θέση να παρέχουν παράλληλη επεξεργασία δεδομένων σε πολύ μεγάλο βαθμό, εξόρυξη μεγάλων δεδομένων, μηχανική εκμάθηση, κ.λ.π. (Yu et al. (2018)).

3.7 Ενσωμάτωση Διαδικτύου των Πραγμάτων και Υπολογιστικής Άκρου

Τα τελευταία χρόνια το IoT και η Edge Computing εξελίσσονται, ανεξάρτητα το ένα από το άλλο, με ταχύτατους ρυθμούς. Παρ' ότι είναι ανεξάρτητα, η υποδομή Edge Computing μπορεί να βοηθήσει στην επίλυση κρίσιμων ζητημάτων του IoT και παράλληλα να βελτιώσει την απόδοση του. Έτσι, έχουν γίνει έρευνες σχετικές με την ανάγκη για ενσωμάτωση τους. Στον Πίνακα 1 καταγράφονται κάποια χαρακτηριστικά που έχει το IoT, το Edge και το Cloud, παρατηρώντας κάποια κοινά μεταξύ IoT-Edge.

Πίνακας 3.7-1: Χαρακτηριστικά IoT Edge Cloud Computing (Yu et al. (2018))

	IoT	Edge	Cloud
Ανάθεση	Κατανεμημένη	Κατανεμημένη	Συγκεντρωτική
Στοιχεία	Φυσικές συσκευές	Κόμβοι άκρου	Εικονικοί πόροι
Υπολογιστική	Περιορισμένη	Περιορισμένη	Απεριόριστη
Αποθήκευση	Μικρή	Περιορισμένη	Απεριόριστη
Απόκριση	-	Γρήγορη	Αργή
Μεγάλα δεδομένα	Πηγή	Επεξεργασία	Επεξεργασία

Γενικά, το IoT μπορεί να επωφεληθεί και από το Edge Computing και από το Cloud Computing, λόγω των χαρακτηριστικών που έχουν οι δύο δομές (π.χ. υψηλή υπολογιστική και αποθηκευτική ικανότητα). Παρ' όλα αυτά, το Edge έχει περαιτέρω πλεονεκτήματα συγκριτικά με το Cloud όσον αφορά το IoT, παρά την περιορισμένη υπολογιστική και αποθηκευτική του ικανότητα. Η Edge Computing προσφέρει ικανοποιητική υπολογιστική ικανότητα, αρκετό αποθηκευτικό χώρο και σύντομο χρόνο απόκρισης, στοιχεία ικανοποιητικά σχετικά με τις απαιτήσεις διαφόρων εφαρμογών IoT. Από την μεριά του Edge, μπορεί επίσης να υπάρξει ωφέλεια από το IoT, επεκτείνοντας

την δομή του και δημιουργώντας καταναμημένους και δυναμικούς κόμβους Edge. Ως κόμβοι Edge χρησιμοποιούνται συσκευές IoT ή συσκευές με μια σχετική υπολογιστική ικανότητα για την παροχή υπηρεσιών. Έχουν γίνει πολλές ερευνητικές προσπάθειες με στόχο την εκμετάλλευση του Cloud Computing για να υποστηρίξει το IoT, αλλά σε αρκετές περιπτώσεις, η υπολογιστική άκρου μπορεί να προσφέρει ανταγωνιστικότερες επιδόσεις (Yu et. al. (2018)).

3.8 Πλεονεκτήματα Edge Computing

Η αρχιτεκτονική των τριών επιπέδων της Εικόνας 3.6-2 που περιγράφηκαν παραπάνω (front-end, near-end, far-end) προσφέρει τα παρακάτω πλεονεκτήματα έναντι του Cloud.

- **Μικρότερη καθυστέρηση κατά την μετάδοση**

Η καθυστέρηση μιας εφαρμογής (latency) είναι το αποτέλεσμα της υπολογιστικής καθυστέρησης (computing latency) και της καθυστέρησης μετάδοσης (transmission latency). Εφόσον οι αισθητήρες που χρησιμοποιούνται στο IoT είναι συνήθως συσκευές με περιορισμένη έως μηδενική υπολογιστική ισχύ ενώ οι διακομιστές των Cloud παρέχουν γρήγορη επεξεργασία δεδομένων λόγω της υψηλής υπολογιστικής τους ισχύος, καθίσταται σαφές ότι η μετάδοση δεδομένων μεταξύ των τελικών συσκευών και των διακομιστών Cloud προκαλεί σημαντική αύξηση της καθυστέρησης μετάδοσης. Οι κόμβοι Edge μπορούν να επεξεργάζονται τα δεδομένα, να τα φιλτράρουν, να λαμβάνουν κάποια απόφαση (σε περίπτωση ενεργοποίησης κάποιου ενεργοποιητή) και να αποστέλλουν μόνο σημαντικές πληροφορίες που πρέπει να μεταδίδονται στο κεντρικό Cloud. Έτσι μειώνεται η καθυστέρηση μετάδοσης και βελτιώνεται η υπολογιστική καθυστέρηση.

- **Αύξηση εύρους ζώνης**

Καθώς οι συνδεδεμένες συσκευές στο IoT όλο και αυξάνονται, η ποσότητα των δεδομένων που παράγονται από αυτές είναι αρκετά μεγάλη. Η μετάδοση μεγάλου όγκου δεδομένων στο Cloud επιβαρύνει το εύρος ζώνης της επικοινωνίας στα ασύρματα δίκτυα, το οποίο οδηγεί σε καθυστερήσεις, απώλεια πακέτων και υποβαθμισμένες υπηρεσίες (Deng et al. (2016)). Η Υπολογιστική Ομίχλης έχει μια αποκεντρωμένη αρχιτεκτονική, φέρνοντας τους υπολογιστικούς πόρους και τις υπηρεσίες εφαρμογών στην άκρη του δικτύου όπου δημιουργούνται και καταναλώνονται τα δεδομένα (Xiao και Zhu (2017)).

- **Ελαχιστοποίηση κατανάλωσης ενέργειας**

Η ταχεία αύξηση του αριθμού των αντικειμένων και των χρηστών που συνδέονται στο Διαδίκτυο σχετίζεται με την απαίτηση για μεγιστοποίηση της ενεργειακής απόδοσης. Οι εργασίες ανατίθενται από τις τελικές συσκευές σε συσκευές Edge που δεν βρίσκονται μακριά, όπως τα κέντρα δεδομένων Cloud. Αυτό το γεγονός συμβάλλει στη μείωση της κατανάλωσης ενέργειας στις τελικές συσκευές, στις κεντρικές υποδομές υπολογιστών και σε κόμβους δικτύου μεταξύ Edge και κεντρικού Cloud καθώς δεν δαπανάται μεγάλο ποσοστό ενέργειας για την μετάδοση στα κέντρα δεδομένων (Tang et al. (2017))

- **Αποθήκευση**

Στην Υπολογιστική Ομίχλης οι κόμβοι Edge είναι υπεύθυνοι για την παροχή υπηρεσιών αποθήκευσης παρέχοντας ευέλικτη αποθήκευση, υπολογιστικούς πόρους και υπηρεσίες που μπορούν να μοιραστούν στο δίκτυο, μειώνοντας έτσι την ανάγκη ανάπτυξης ειδικών διακομιστών και κέντρων δεδομένων (Naranjo et al. (2016)).

- **Αποφυγή υπολογιστικής συμφόρησης**

Στην άκρη του δικτύου, ο φόρτος ανάλυσης δεδομένων παραλληλοποιείται σε ένα μεγάλο αριθμό υπολογιστικών μονάδων. Κάθε μονάδα πρέπει να εκτελεί μόνο ελαφρές υπολογιστικές εργασίες, ωστόσο, οι υπολογιστικές δυνατότητες που διαθέτουν παρέχουν υπολογιστική ισχύ υψηλής απόδοσης. Επιπλέον, η απόδοση (throughput) και το φορτίο μεταξύ όλων των υπολογιστικών κόμβων μπορεί να εξισορροπηθεί εύκολα αποφεύγοντας πιθανά σημεία υπολογιστικής συμφόρησης (Tang et al. (2017)).

3.9 Ασφάλεια και ιδιωτικότητα στο Άκρο του δικτύου

Ως «ευαίσθητοι στόχοι» που εμπλέκονται σε πολλούς τομείς, η ασφάλεια και η ιδιωτικότητα αποτελούν κρίσιμα ζητήματα που απαιτούν προσεκτική εξέταση. Μέσα από την συνεργασία IoT και Edge Computing προκύπτουν κάποια σημαντικά ζητήματα. Η Υπολογιστική Άκρου επικεντρώνεται γύρω από έναν πολύπλοκο συνδυασμό πολλαπλών και διαφόρων τεχνολογιών (συστήματα peer to peer, ασύρματα δίκτυα, εικονικοποίηση κ.α) και απαιτεί την υιοθέτηση ενός ολοκληρωμένου συστήματος για την προστασία και την διαχείριση της εκάστοτε τεχνολογικής πλατφόρμας και του συστήματος γενικά.

Απειλές ασφαλείας μπορούν να προκύψουν σε οποιαδήποτε από τα στάδια επεξεργασίας δεδομένων, συμπεριλαμβανομένης της συλλογής και φιλτραρίσματος

δεδομένων, αναπαράσταση, ενσωμάτωση δεδομένων, μοντελοποίηση και ερμηνεία. Επιπλέον, η εκθετική αύξηση του εύρους δεδομένων δημιουργεί δυσκολίες στην εγγύηση της ασφάλειας κρίσιμων δεδομένων). Ένα κρίσιμο ζήτημα είναι η ασφάλιση των συνδέσεων ενός τεράστιου αριθμού ετερογενών συσκευών IoT. Στην Υπολογιστική Ομίχλη, οι κατανεμημένες συσκευές IoT έχουν ελάχιστες δυνατότητες προστασίας και επίσης δεν διαθέτουν ολική γνώση του δικτύου, επομένως είναι πιο ευάλωτες σε επιθέσεις. Σε αυτή την περίπτωση, οι κόμβοι ομίχλης μπορούν να παίξουν το ρόλο των διακομιστών μεσολάβησης για να διαχειριστούν τις λειτουργίες ασφαλείας ορισμένων πόρων συσκευών IoT (Mukherjee et al. (2017)). Ωστόσο, η κοινή χρήση δεδομένων μεταξύ των τελικών χρηστών στο περιβάλλον ομίχλης αυξάνει την ευαλωτότητα των παραγόμενων δεδομένων.

Για την διατήρηση της ιδιωτικότητας των τελικών χρηστών είναι σημαντικό να παρέχονται αποτελεσματικές τεχνικές προσθήκης ψευδωνύμων, αλγόριθμοι διατήρησης της ιδιωτικότητας, κρυπτογράφηση, ανωνυμία και προσωρινή ταυτοποίηση. Πριν την αποστολή των δεδομένων στο Cloud, οι κόμβοι ομίχλης μπορούν να χρησιμοποιηθούν για τον έλεγχο και την αποθήκευση κρίσιμων δεδομένων που αφορούν την ιδιωτικότητα, όμως η κατανεμημένη αποθήκευση που παρέχει η ομίχλη καθώς και οι περιορισμένες δυνατότητες ορισμένων κόμβων δημιουργούν δυσκολίες στην χρήση των υπάρχουσών τεχνικών διατήρησης της ιδιωτικότητας (Tsai et al. (2015)).

Έτσι είναι σημαντικό να σχεδιαστούν μηχανισμοί που θα διασφαλίζουν ότι τα δεδομένα που δημιουργούνται από το IoT μεταδίδονται με αξιόπιστο τρόπο και δεν παραβιάζεται η επικοινωνία των κόμβων ομίχλης με τις συσκευές IoT. Επίσης, θα πρέπει να αναπτυχθούν ελαφριοί και γρήγοροι αλγόριθμοι προστασίας της ασφάλειας και της ιδιωτικότητας οι οποίοι να μπορούν να αντιμετωπίσουν το δυναμικό και ετερογενή περιβάλλον του συνδυασμού υπολογιστικής ομίχλης-IoT. (Yu et al. (2018)).

3.10 Υπολογιστική Άκρου στα συνδεδεμένα οχήματα

Ιστορικά, η υιοθέτηση τεχνολογιών Υπολογιστικού Νέφους και λογισμικού χρησιμοποιείται από τις βιομηχανίες εδώ και αρκετά χρόνια, όμως η τηλεματική των οχημάτων όσον αφορά την αυτοκινητοβιομηχανία έγινε μια από τις κορυφαίες περιπτώσεις χρήσης γύρω στο 2008.

Τα συνδεδεμένα οχήματα συνεχίζουν να εξελίσσονται με εκθετικούς ρυθμούς, με την ανάπτυξη των επικοινωνιών όχημα-προς-όχημα (V2V) και όχημα-προς-κάθε

πράγμα (V2X) δημιουργώντας τεράστιους όγκους δεδομένων. Έτσι γεννάται το ερώτημα πως θα διαχειριστεί, επεξεργαστεί και αναλυθεί αυτός ο μεγάλος όγκος δεδομένων λαμβάνοντας κρίσιμες αποφάσεις γρήγορα και αποδοτικά; Μια ομάδα παγκόσμιων βιομηχανιών έχει δημιουργήσει το Automotive Edge Computing Consortium (AECC) το οποίο ασχολείται με την εύρεση βέλτιστων πρακτικών για την σύγκλιση μεταξύ οχήματος και υπολογιστικού περιβάλλοντος. Η Υπολογιστική Άκρου προσφέρει αρκετά σενάρια χρήσης στα υποσυστήματα των οχημάτων από τον αισθητήρα, τις εφαρμογές έως την εμπειρία που λαμβάνει ο τελικός χρήστης. Μερικές εφαρμογές περιγράφονται παρακάτω:

- **Προληπτική συντήρηση**

Με την υποστήριξη ορισμένων προγνωστικών αναλυτικών μοντέλων, η Υπολογιστική Άκρου μπορεί να παρακολουθεί διάφορες παραμέτρους του οχήματος, όπως θερμοκρασία κινητήρα, χιλιόμετρα, πίεση ελαστικών, επιτάχυνση και ταχύτητα. Το μοντέλο ανάλυσης που «τρέχει» στη συσκευή Edge θα μπορεί να προβλέψει εάν κάποιο εξάρτημα ενδέχεται να αστοχήσει και να ειδοποιήσει τον ιδιοκτήτη του οχήματος. Για παράδειγμα αν οι τιμές του αισθητήρα πίεσης δεν είναι οι αποδεκτές, η συσκευή Edge θα ειδοποιεί άμεσα τον ιδιοκτήτη του οχήματος για αντικατάσταση του αισθητήρα πριν προλάβει να εμφανιστεί βλάβη.

- **Μεταφόρτωση δεδομένων αισθητήρων**

Ένα τυπικό όχημα περιέχει δεκάδες αισθητήρες. Παρόλο που όλα τα δεδομένα που παράγονται υποβάλλονται σε επεξεργασία από το όχημα, διάφορες εφαρμογές εντός του οχήματος απαιτούν τη μεταφορά δεδομένων στο Cloud. Με την Υπολογιστική Άκρου, τα δεδομένα που προωθούνται στο Cloud θα μπορούν να διαχειριστούν πιο έξυπνα. Η συσκευή άκρου μπορεί να επεξεργαστεί και να αναλύσει τα περισσότερα δεδομένα στην άκρη του δικτύου και να μεταφορτώσει μόνο επιλεγμένα μη ευαίσθητα δεδομένα στο Cloud. Αυτό μειώνει το κόστος μετάδοσης δεδομένων και προστατεύει επίσης τα ευαίσθητα δεδομένα να μην εξέλθουν από το όχημα.

- **Ασφάλεια και ταυτοποίηση**

Κατά τον έλεγχο ταυτότητας πολλαπλών επιπέδων η συσκευή Edge μπορεί να διαχειρίζεται πολλαπλούς αισθητήρες όπως κάμερες και αισθητήρες εγγύτητας. Τα δεδομένα από αυτές τις συσκευές μπορούν να χρησιμοποιηθούν για να επιτρέψουν τον έλεγχο ταυτότητας πολλαπλών επιπέδων όπου: η κάμερα θα εκτελεί αναγνώριση προσώπου, ένας αισθητήρας Bluetooth θα ανιχνεύει την εγγύτητα του τηλεφώνου του

οδηγού και η συσκευή WiFi θα ανιχνεύει πιθανή επίθεση πλαστογράφησης (Spoofing attack) στο σύστημα ασφαλείας του οχήματος.

Συνοψίζοντας, το Edge μπορεί να παρέχει σημαντική υποστήριξη στον τομέα των μεταφορών παρέχοντας την δυνατότητα τα δεδομένα που παράγονται να μην ανταλλάσσονται απευθείας μεταξύ Edge και Cloud, δημιουργώντας καθυστερήσεις στις υποδομές του δικτύου. Έτσι η υποστήριξη παρέχεται με λειτουργικές εφαρμογές σε ένα ευρύ φάσμα όπως η κατάσταση του οχήματος, η κατάσταση του οδικού δικτύου, η ασφάλεια και οι οδηγικές συμπεριφορές των χρηστών. Η μεγαλύτερη πρόκληση που έχει να αντιμετωπίσει, είναι η αποτελεσματική επεξεργασία και αποστολή των δεδομένων που παράγονται μέσω των υπάρχοντων δικτύων επικοινωνίας, ακόμα και αν το δίκτυο 5G είναι διαθέσιμο παντού. Επίσης, η πρόκληση δεν περιορίζεται μόνο στην διαχείριση του μεγάλου όγκου δεδομένων που δημιουργούνται από τα συνδεδεμένα οχήματα, αλλά και στη διασφάλιση της ασφάλειας, της αξιοπιστίας και της κλιμάκωσης των δεδομένων.

4 Ανάπτυξη πειραματικού περιβάλλοντος

Σκοπός της πειραματικής ανάλυσης είναι η διερεύνηση του αντίκτυπου της εγκατάστασης κόμβου Edge Cloud στο όχημα, σε αντίθεση με την απουσία χρήσης του συγκεκριμένου κόμβου, όσον αφορά την απόδοση του δικτύου. Η πειραματική ανάλυση υλοποιείται μέσω παρακολούθησης της καθυστέρησης μετάδοσης και της κατανάλωσης εύρους ζώνης κατά την αποστολή δεδομένων, ορισμένων αισθητήρων, από την θύρα OBD του οχήματος. Υλοποιούνται δύο πειραματικά σενάρια, όπου στο πρώτο συλλέγονται δεδομένα και αποστέλλονται απευθείας προς έναν απομακρυσμένο διακομιστή (Core Cloud), ενώ στο δεύτερο τα δεδομένα συλλέγονται με τον ίδιο τρόπο και αποστέλλονται προς τον ίδιο απομακρυσμένο διακομιστή μέσω μιας Edge συσκευής η οποία εγκαταστάθηκε στο όχημα. Για να πραγματοποιηθεί το πείραμα χρησιμοποιήθηκαν τεχνολογίες hardware, λογισμικού, πόρων αποθήκευσης και πρωτόκολλα επικοινωνίας.

Το περιβάλλον του πρώτου σεναρίου αποτελείται από το όχημα το οποίο διαθέτει θύρα OBDII και δημιουργεί δεδομένα από τους αισθητήρες του κατά την κίνηση σε ελεγχόμενο χώρο όπου εκπέμπεται δίκτυο 4G. Επίσης στον ίδιο χώρο, μακριά από το όχημα, υπάρχει σε συγκεκριμένο σημείο φορητή υπολογιστική μονάδα η οποία είναι συνδεδεμένη στο παρεχόμενο δίκτυο. Για την καταγραφή των δεδομένων από τους

αισθητήρες χρησιμοποιήθηκε η εφαρμογή Torque, που εκτός από τις δυναμικές μετρήσεις που παρέχει, μεταφορτώνει τα καταγεγραμμένα δεδομένα σε ένα webserver, αξιοποιώντας το διαθέσιμο δίκτυο. Η εφαρμογή Torque είναι εγκατεστημένη σε smartphone που βρίσκεται εντός του οχήματος. Την ίδια στιγμή στον φορητό υπολογιστή «τρέχει» το λογισμικό node-RED, το οποίο λαμβάνει τα δεδομένα και προγραμματίστηκε να τα αποστέλλει στο Azure Cloud.

Το περιβάλλον του δευτέρου σεναρίου αποτελείται από το ίδιο όχημα που μετακινείται στον ίδιο χώρο. Σε αυτή την περίπτωση το δίκτυο παρέχεται από συσκευή smartphone που λειτουργεί ως Access Point. Στο όχημα έχει εγκατασταθεί κόμβος Edge και συνδέεται στο παρεχόμενο 4G δίκτυο, ενώ εδώ απουσιάζει ο φορητός υπολογιστής. Η εφαρμογή Torque λαμβάνει τα δεδομένα των αισθητήρων και σε συνεργασία με την συσκευή Edge, τα δεδομένα αποστέλλονται στην ίδια πλατφόρμα Azure, μέσω του node-RED. Σε αυτό το σενάριο, στη συσκευή Edge έχει εγκατασταθεί το λογισμικό node-RED και χρησιμοποιείται για το φιλτράρισμα των δεδομένων, έτσι ώστε να μην μεταφέρονται όλα όσα μετρώνται από την εφαρμογή Torque αλλά μόνο τα κρίσιμα.

Οι τεχνολογίες hardware και software που αξιοποιήθηκαν περιγράφονται αναλυτικότερα στα υποκεφάλαια 4.1 και 4.2. Στη συνέχεια, στο υποκεφάλαιο 4.3 περιγράφεται το περιβάλλον των πειραματικών σεναρίων, ενώ στο 4.4 αναλύεται ο τρόπος ανάπτυξης των εργαλείων hardware και software που χρησιμοποιήθηκαν. Τέλος, στο 4.5 αναφέρεται ο τρόπος με τον οποίο γίνεται η παρακολούθηση της καθυστέρησης μετάδοσης και της κατανάλωσης εύρους ζώνης.

4.1 Τεχνολογίες Hardware

Σε αυτό το υποκεφάλαιο παρουσιάζονται οι τεχνολογίες υλικού που χρησιμοποιήθηκαν για την υλοποίηση της πειραματικής ανάλυσης και περιγράφονται τα χαρακτηριστικά τους. Αρχικά παρουσιάζεται το ενσωματωμένο υπολογιστικό σύστημα Raspberry Pi το οποίο χρησιμοποιήθηκε ως συσκευή Edge, στη συνέχεια ακολουθεί η περιγραφή της διεπαφής OBD και του πρωτοκόλλου που χρησιμοποιήθηκε για την σύνδεση της διαγνωστικής συσκευής με την εφαρμογή η οποία συλλέγει τα δεδομένα. Τέλος, αναλύονται τα πρωτόκολλα που σχετίζονται με την επικοινωνία πελάτη – εξυπηρετητή.

4.1.1 Raspberry Pi

Το Raspberry Pi είναι μια σειρά μικρών υπολογιστών μονής πλακέτας που αναπτύχθηκαν από το ίδρυμα Raspberry Pi του Ηνωμένου Βασιλείου με σκοπό την προώθηση της διδασκαλίας της βασικής επιστήμης των υπολογιστών σε σχολεία και σε αναπτυσσόμενες χώρες (“Raspberry Pi”, 2020). Έχουν κυκλοφορήσει αρκετές εκδόσεις με διαφορές σε επεξεργαστική ισχύ, μέγεθος RAM, αριθμό διαθέσιμων ακίδων σύνδεσης εξαρτημάτων, αριθμό θυρών USB ενώ οι νεότερες πλακέτες διαθέτουν ενσωματωμένα Bluetooth και WiFi.

Στο πείραμα έγινε χρήση της έκδοσης Raspberry 4 Model B το οποίο έχει τα ακόλουθα χαρακτηριστικά:

- Επεξεργαστής: Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- RAM 2GB
- Wi-Fi 2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless
- LAN, Bluetooth 5.0, BLE
- 4 Θύρες USB
- 40 επαφές GPIO
- 2 micro HDMI ports
- Διεπαφή κάμερας και οθόνης
- Θύρα κάρτας microSD λειτουργικού συστήματος

Λόγω της ύπαρξης του επεξεργαστή ARMv8 το Raspberry μπορεί να «τρέξει» αρκετές ARM GNU/LINUX διανομές με κυριότερο και προτεινόμενο από την εταιρεία κατασκευής, το λειτουργικό σύστημα Raspbian αλλά και άλλες διανομές όπως το Ubuntu Core, το LibreELEC, το RISC κ.α.

4.1.2 Διεπαφή OBDII

Ο όρος On-Board Diagnostics (OBD) ανήκει στην ορολογία της αυτοκινητοβιομηχανίας και αναφέρεται στην δυνατότητα αυτοδιάγνωσης και αναφοράς βλαβών του οχήματος. Η διεπαφή OBD δίνει στον ιδιοκτήτη και στον τεχνικό συντήρησης πρόσβαση στην κατάσταση των διάφορων υποσυστημάτων του οχήματος. Η ποσότητα των διαθέσιμων πληροφοριών μέσω του OBD διαφοροποιούνται από την πρώτη εμφάνιση του στις αρχές του 1980. Οι πρώτες εκδόσεις ενεργοποιούσαν μια ενδεικτική λυχνία στο χειριστήριο του οχήματος, δεν παρείχαν όμως καμία πληροφορία

σχετικά με το υποσύστημα που εμφανίστηκε η βλάβη. Τα σύγχρονα OBD χρησιμοποιούν μια τυποποιημένη ψηφιακή θύρα και παρέχουν δεδομένα (από αισθητήρες και ενεργοποιητές) σε πραγματικό χρόνο, επιπρόσθετα σε μια τυποποιημένη σειρά από Διαγνωστικούς Κωδικούς Σφαλμάτων (DTC) οι οποίοι παρέχουν ταχύτατο εντοπισμό βλαβών (“On-Board diagnostics”, 2020). Στο πείραμα της εργασίας χρησιμοποιείται η έκδοση OBDII η οποία είναι μια βελτίωση έναντι του OBDI τόσο στην ικανότητα όσο και στην τυποποίηση. Το πρότυπο OBDII καθορίζει τον τύπο του διαγνωστικού συνδέσμου και το pinout του, τα διαθέσιμα πρωτόκολλα ηλεκτρικής σηματοδότησης και τη μορφή ανταλλαγής μηνυμάτων. Παρέχει επίσης έναν κατάλογο παραμέτρων του οχήματος, για παρακολούθηση και κωδικοποίηση των δεδομένων κάθε παραμέτρου (“On-Board diagnostics”, 2020).

➤ Θύρα OBDII

Η θύρα OBDII (Εικόνα 4.1-1) στην πλειοψηφία των οχημάτων διαθέτει 16 επαφές οι οποίες ανάλογα με τον κατασκευαστή χρησιμοποιούνται για την υλοποίηση ιδιωτικών πρωτοκόλλων επικοινωνίας ή για πρόσβαση σε ιδιόκτητα δίκτυα κατασκευαστών. Η ανίχνευση του πρωτοκόλλου που υλοποιείται σε ένα όχημα μπορεί να εξακριβωθεί από την ύπαρξη ή μη συγκεκριμένων επαφών στην θύρα OBDII. Τα κυριότερα πρωτόκολλα είναι τα εξής: ISO 15765-4, ISO 14230-4, ISO 9141-2, SAE J1850 PWM και VPW, ISO 11898, ISO 15765, SAE J1939.



Εικόνα 4.1-1: Θύρα OBD2 σε NISSAN NV300

Τα συστήματα OBD υποστηρίζουν 10 διαφορετικές λειτουργίες (modes) του πρωτοκόλλου επικοινωνίας. Σε νεότερα οχήματα όπου διάφορες ενέργειες διεξάγονται

από ηλεκτρονικά υποσυστήματα υπάρχει η περίπτωση να υποστηρίζονται περισσότερες λειτουργίες. Παρακάτω περιγράφονται οι 10 λειτουργίες του OBD:

Λειτουργία 1^η: Προσδιορίζει ποιες πληροφορίες των συστημάτων μετάδοσης κίνησης είναι διαθέσιμες στη διαγνωστική συσκευή.

Λειτουργία 2^η: Επιστρέφει δεδομένα freeze frame (το freeze frame είναι μια αναφορά που περιέχει περιγραφή, κώδικα DTC και την κατάσταση του κώδικα OBDII στη μορφή στιγμιότυπου οθόνης) που σχετίζονται με ένα κωδικό σφάλματος.

Λειτουργία 3^η: Παραθέτει τους αποθηκευμένους κωδικούς βλάβης που έχουν επιβεβαιωθεί σχετικά με τις εκπομπές καυσαερίων. Εμφανίζει ακριβείς αριθμητικούς, τετραψήφιους κωδικούς που προσδιορίζουν τα σφάλματα.

Λειτουργία 4^η: Χρησιμοποιείται για την εκκαθάριση διαγνωστικών πληροφοριών που σχετίζονται με τις εκπομπές καυσαερίων. Αυτό περιλαμβάνει την εκκαθάριση αποθηκευμένων DTC που εκκρεμούν ή έχουν επιβεβαιωθεί καθώς και δεδομένα freeze frame.

Λειτουργία 5^η: εμφανίζει τα αποτελέσματα των δοκιμών που συγκεντρώθηκαν σχετικά με τον αισθητήρα οξυγόνου.

Λειτουργία 6^η: Σχετίζεται με την αίτηση για τα αποτελέσματα των δοκιμών παρακολούθησης του συστήματος η οποία μπορεί να είναι είτε συνεχής είτε μη-συνεχής.

Λειτουργία 7^η: Αφορά ένα αίτημα για κωδικούς βλάβης που σχετίζονται με εκπομπές καυσαερίων που εντοπίστηκαν κατά τη διάρκεια του τρέχοντος ή του τελευταίου ολοκληρωμένου κύκλου οδήγησης. Η λειτουργία αυτή χρησιμοποιείται από τεχνικούς μετά από επισκευή του οχήματος και μετά την εκκαθάριση των διαγνωστικών πληροφοριών έτσι ώστε να είναι σε θέση να παρατηρήσει τα αποτελέσματα των δοκιμών μετά από έναν μόνο κύκλο οδήγησης για να προσδιοριστεί εάν η επισκευή έχει επιδιορθώσει το πρόβλημα.

Λειτουργία 8^η: Η λειτουργία αυτή επιτρέπει στην διαγνωστική συσκευή να ελέγχει τις διαδικασίες ενός ενσωματωμένου συστήματος ή εξαρτήματος.

Λειτουργία 9^η: Χρησιμοποιείται για την ανάκτηση πληροφοριών του οχήματος όπως αναγνωριστικό οχήματος, αναγνωριστικό του λογισμικού που εγκαταστάθηκε στην ηλεκτρονική μονάδα ελέγχου, αν ο κινητήρας διαθέτει καταλύτη ανάλογα με τον τύπο καυσίμου κ.α.

Λειτουργία 10^η: Καταγράφει τους μόνιμους διαγνωστικούς κωδικούς βλάβης σχετικά με τις εκπομπές καυσαερίων που έχουν αποθηκευτεί (“On-Board diagnostics”, 2020).

4.1.3 Συσκευή διάγνωσης ELM327

Η συσκευή ELM327 (Εικόνα 4.1-2) είναι ένας μικροελεγκτής που κατασκευάζεται από την ELM Electronics και εισάγεται στην θύρα OBD των οχημάτων. Μέσω της θύρας OBDII επικοινωνεί με την ηλεκτρονική μονάδα ελέγχου και παρέχει στον χρήστη διαγνωστικό έλεγχο σφαλμάτων και τιμές δυναμικών μετρήσεων αισθητήρων όπως θερμοκρασία ψυκτικού υγρού, τάση φόρτισης μπαταρίας κ.α. Η συσκευή επικοινωνεί ασύρματα (Bluetooth, WiFi) ή ενσύρματα με PC ή smartphone/tablet. Στο πείραμα χρησιμοποιήθηκε συσκευή με Bluetooth. Επιλέχθηκε στην εργασία διότι υποστηρίζει τα περισσότερα SAE και ISO πρωτόκολλα επικοινωνίας καθώς και εντολές AT. Η ELM327 αναλαμβάνει την ανίχνευση του πρωτοκόλλου και να το ερμηνεύσει. Από την άλλη πλευρά, υπάρχει ένα smartphone με λειτουργικό σύστημα Android που επικοινωνεί με την ELM327 μέσω Bluetooth και ανταλλάσσουν μεταξύ τους σειριακά δεδομένα. Όσον αφορά το δεύτερο κομμάτι, δηλαδή το λογισμικό, η όλη εργασία γίνεται από την εφαρμογή που έχει αποθηκευτεί στο smartphone, η οποία έχει ρυθμιστεί να κάνει κάποιες διαδικασίες.



Εικόνα 4.1-2: OBD2 Adapter ELM327

4.1.4 Τεχνολογία Bluetooth

Το Bluetooth είναι ένα βιομηχανικό πρότυπο με χρήση στα ασύρματα προσωπικά δίκτυα υπολογιστών (Wireless Personal Area Networks, WPAN). Πρόκειται για μια ασύρματη τηλεπικοινωνιακή τεχνολογία διαθέσιμη σε κοντινές αποστάσεις, η οποία είναι σε θέση να μεταδίδει σήματα με μικρο-κύματα σε ψηφιακές συσκευές. Επομένως το Bluetooth είναι ένα πρωτόκολλο το οποίο παρέχει προτυποποιημένη, ασύρματη

επικοινωνία ανάμεσα σε PDA's, κινητά τηλέφωνα, φορητούς υπολογιστές, , εκτυπωτές κ.α. Για να επιτευχτεί αυτό χρησιμοποιεί μια ασφαλή, φθηνή και παγκοσμίως διαθέσιμη ραδιοσυχνότητα μικρής εμβέλειας. Επίσης, λειτουργεί στο φάσμα συχνοτήτων των 2,4 GHz, έτσι ώστε οι συσκευές στις οποίες είναι ενσωματωμένο να λειτουργούν χωρίς προβλήματα σε οποιοδήποτε σημείο του πλανήτη. Μέσω της σύνδεσης δυο συσκευών μέσω Bluetooth καθίσταται δυνατή η ανταλλαγή δεδομένων, η σύνδεση στο Internet κ.α. Όσον αφορά την δομή του, η βασική δομική του μονάδα είναι το piconet, στο οποίο όλοι οι κόμβοι που λαμβάνουν μέρος (έως 7 συσκευές slaves) λαμβάνουν έναν κοινό κώδικα διασποράς και ελέγχονται από έναν master που είναι ίδιος για όλους. Ο master ελέγχει οποιαδήποτε μεταβολή δικτυακής τοπολογίας (εισαγωγή-αποχώρηση κόμβου), ενώ το piconet δημιουργείται από ένα κόμβο που θέλει να γίνει master ("Bluetooth", 2020). Στην συγκεκριμένη εργασία, η τεχνολογία Bluetooth χρησιμοποιήθηκε για την επικοινωνία του ELM327 με την εφαρμογή που συλλέγει τα δεδομένα από την θύρα OBDII σε πραγματικό χρόνο.

4.1.5 Ασύρματα τοπικά δίκτυα (Tethering, 4G, LTE)

Με το όρο Ασύρματο Τοπικό Δίκτυο (WLAN) νοείται ένα ασύρματο δίκτυο υπολογιστών όπου είναι συνδεδεμένοι δύο ή περισσότερες συσκευές, κάνοντας χρήση μιας ασύρματης μεθόδου διανομής εντός μιας περιοχής περιορισμένης έκτασης. Παρέχει στους χρήστες την δυνατότητα να παραμένουν συνδεδεμένοι στο δίκτυο ενώ μετακινούνται εντός της τοπικής περιοχής κάλυψης καθώς και τη δυνατότητα σύνδεση στο ευρύτερο Διαδίκτυο. Τα πιο πρόσφατα WLANs βασίζονται στο πρότυπο IEEE 802.11 και συναντώνται ευρέως ως WiFi ("Wireless LAN", 2020). Κάθε συσκευή που χρησιμοποιεί WiFi μπορεί να αποκτήσει πρόσβαση σε κάποιο πόρο του διαδικτύου μέσω ενός σημείου πρόσβασης ασυρμάτου δικτύου (Access Point). Τα πρότυπα IEEE 802.11a IEEE 802.11b που δημιουργήθηκαν αρχικά δεν μπόρεσαν να καλύψουν τις ζητούμενες απαιτήσεις και έτσι εν συνεχεία δημιουργήθηκαν τα πρότυπα IEEE 802.11g, IEEE 802.11n, IEEE 802.11ac και IEEE 802.11ax τα οποία προσέφεραν καλύτερη μετάδοση, κατανάλωση ενέργειας και αξιοπιστία. Το πρότυπο IEEE 802.11n λειτουργεί σε συχνοτήτες με εύρος ζώνης 2.4 - 5 GHz και μπορεί να μεταφέρει δεδομένα με ταχύτητα έως 150 Mbps, ενώ το IEEE 802.11ax λειτουργεί με εύρος ζώνης 2.4 - 5 GHz και ταχύτητα έως και 9600 Mbps ("Wi-Fi", 2020). Στο πείραμα της εργασίας το Raspberry

Pi 4 έχει ενσωματωμένο WiFi και υποστηρίζει τα πρότυπα IEEE 802.11b/g/n/ac για την επικοινωνία με το διαδίκτυο.

4.1.6 Λειτουργία Tethering και δίκτυο 4G

Η λειτουργία Tethering επιτρέπει την κοινή χρήση της σύνδεσης Internet μιας κινητής συσκευής (smartphone ή tablet) με κάποια άλλη συσκευή. Η σύνδεση μιας κινητής συσκευής με άλλες συσκευές καθίσταται εφικτή μέσω ασύρματου LAN, μέσω Bluetooth ή μέσω φυσικής σύνδεσης χρησιμοποιώντας καλώδιο. Αν η σύνδεση πραγματοποιείται μέσω WLAN, η λειτουργία θεωρείται ως φορητό σημείο πρόσβασης, το οποίο επιτρέπει στη συσκευή να συμπεριφέρεται ως φορητός δρομολογητής (“Tethering”, 2020). Η κινητή συσκευή που χρησιμοποιήθηκε στην εργασία ως Access Point είναι το smartphone Huawei P9, για να συνδεθεί μέσω του smartphone, το Raspberry στο διαδίκτυο.

Όσον αφορά τον τρόπο επικοινωνίας του Raspberry με το Διαδίκτυο και πιο συγκεκριμένα με την πλατφόρμα Azure IoT της Microsoft, αυτή γίνεται μέσω δικτύου 4G (όπου G - Generation). Το 4G είναι η τέταρτη γενιά της τεχνολογίας ευρυζωνικών κυψελοειδών δικτύων, που ακολούθησε μετά το 3G. Στον τομέα των τηλεπικοινωνιών, ως «γενιά» αναφέρεται μια αλλαγή στη θεμελιώδη αρχή της υπηρεσίας, στην τεχνολογία μετάδοσης, σε υψηλότερους ρυθμούς bit, σε νέες ζώνες συχνοτήτων, σε μεγαλύτερο εύρος ζώνης συχνότητας καναλιών (σε Hertz) και μεγαλύτερη δυνατότητα για πολλαπλές ταυτόχρονες μεταφορές δεδομένων. Ένα σύστημα 4G πρέπει να υποστηρίζει κάποιες απαιτήσεις που ορίζονται από την International Telegraph Union στο IMT Advanced (International Mobile Telecommunications Advanced). Η προδιαγραφή IMT Advanced όρισε τις απαιτήσεις ταχύτητας αιχμής για την υπηρεσία 4G, στα 100Mbps για επικοινωνίες όπου υπάρχει υψηλή κινητικότητα και 1Gbps για επικοινωνίες χαμηλής κινητικότητας. Το 4G δίκτυο λειτουργεί σε συχνότητες 2 – 8 GHz και έτσι η μεταφορά δεδομένων καθίσταται εφικτή με πολύ μεγαλύτερες ταχύτητες σε σχέση με το προηγούμενης γενιάς 3G. Ορισμένες πιθανές και υπάρχουσες εφαρμογές περιλαμβάνουν την τροποποιημένη πρόσβαση στον κινητό ιστό, υπηρεσίες VoIP, υπηρεσίες παιχνιδιών, την κινητή τηλεόραση υψηλής ευκρίνειας, την τηλεδιάσκεψη και την τρισδιάστατη τηλεόραση.

4.2 Τεχνολογίες Software

Στο παρόν υποκεφάλαιο παρουσιάζονται και αναλύονται οι τεχνολογίες λογισμικού που χρησιμοποιήθηκαν στο σενάριο χρήσης. Οι υποενότητες χωρίζονται ως εξής: Αρχικά αναφέρεται το λειτουργικό σύστημα του Raspberry. Στην συνέχεια ακολουθεί η εφαρμογή OBD Torque και το λογισμικό εργαλείο node-RED της IBM. Ακολούθως, γίνεται αναφορά στην υπηρεσία Cloud Azure, στην πλατφόρμα Docker και το πρωτόκολλο μεταφοράς υπερκειμένου.

4.2.1 Λειτουργικό σύστημα *Raspberry Pi*

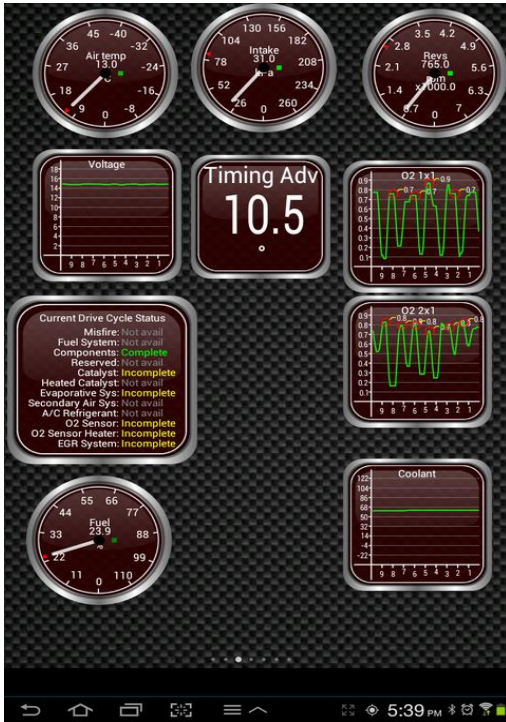
Το Raspberry Pi OS αποτελεί ένα ελεύθερο λειτουργικό σύστημα βασισμένο στο Debian και έχει βελτιστοποιηθεί έτσι ώστε να συνεργάζεται με τον υπολογιστή μονής πλακέτας Raspberry Pi. Παρέχεται επίσημα από το Ίδρυμα Raspberry Pi και δημιουργήθηκε από τους Mike Thompson και Peter Green ως ανεξάρτητο έργο το 2012. Οι παλαιότερες εκδόσεις ονομάζονταν Raspbian, ήταν 32bit και χρησιμοποιούσαν πυρήνα Raspbian. Οι νεότερες εκδόσεις των 64bit δεν κάνουν χρήση του πυρήνα Raspbian και έτσι η ονομασία του λειτουργικού άλλαξε σε Raspberry Pi OS και για τις δυο εκδόσεις 32 και 64 bit. Το Raspberry Pi OS έχει βελτιστοποιηθεί αρκετά ώστε να ταιριάζει στην πλακέτα και να συνεργάζεται αποδοτικά με τον επεξεργαστή ARM (“Raspberry Pi OS”, 2020). Έχει προεγκατεστημένες τις Python, Java, Mathematica και άλλα λογισμικά. Στην πλακέτα του πειράματος έγινε εγκατάσταση της έκδοσης Raspbian Buster με έκδοση kernel 4.19.

4.2.2 Εφαρμογή *Torque*

Η εφαρμογή Torque έχει αναπτυχθεί από τον Ian Hawkins και είναι ένα εργαλείο διάγνωσης σφαλμάτων και παρακολούθησης των τιμών ορισμένων αισθητήρων του αυτοκινήτου σε πραγματικό χρόνο, παρέχοντας τις πληροφορίες αυτές μέσω ενός OBDII reader (όπως είναι το ELM327). Έχει σχεδιαστεί για το λειτουργικό σύστημα Android και παρέχει δυνατότητα σύνδεσης με το ELM327 με Bluetooth ή WiFi. Το γραφικό περιβάλλον της εφαρμογής φαίνεται στην Εικόνα 4.2-1. Επίσης παρέχει τις ακόλουθες δυνατότητες:

- Παρουσίαση των καταγεγραμμένων σφαλμάτων και εκκαθάριση ιστορικού σφαλμάτων.
- Παρουσίαση σφαλμάτων που εκκρεμούν.

- Παρακολούθηση μετρήσεων κάποιων αισθητήρων και αποστολή των καταγεγραμμένων τιμών μέσω e-mail.
- Παρακολούθηση μετρήσεων κάποιων αισθητήρων και μεταφόρτωση των καταγεγραμμένων τιμών σε κάποιο διακομιστή web.



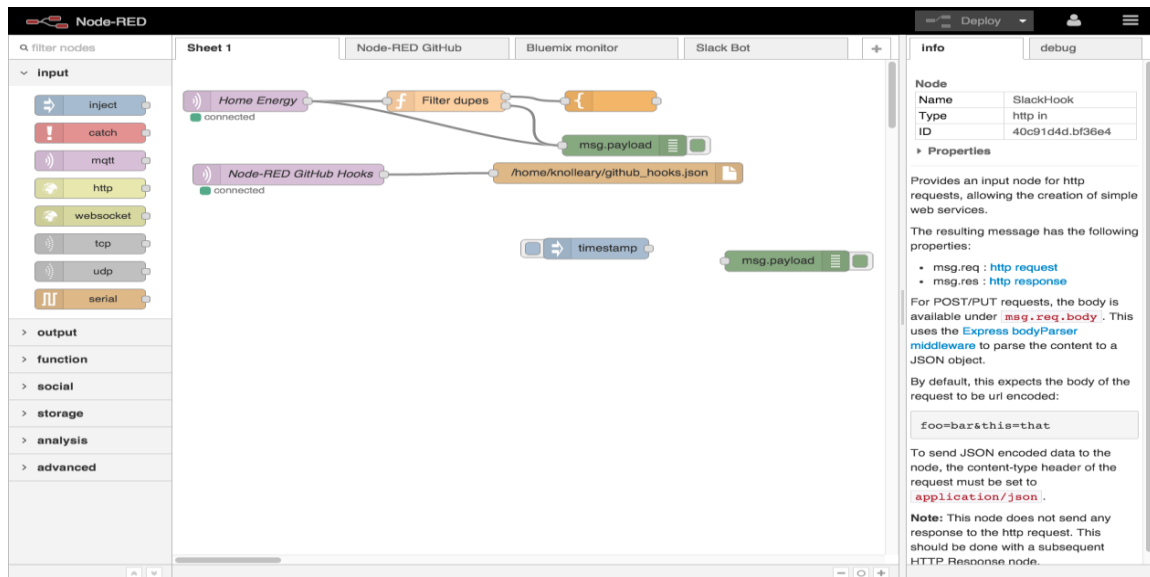
Εικόνα 4.2-1: Γραφικό περιβάλλον εφαρμογής Torque

Η συγκεκριμένη εφαρμογή χρησιμοποιήθηκε στην εργασία α) για την μετατροπή των δεδομένων που λαμβάνονται από την θύρα OBDII σε πραγματικές τιμές αισθητήρων, διότι δεν μπορούν τα δεδομένα που λαμβάνονται απευθείας από την θύρα να αξιοποιηθούν για διάγνωση και β) για την μεταφόρτωση των δεδομένων στον διακομιστή του node-RED.

4.2.3 Node-RED

Το node-RED είναι εργαλείο προγραμματισμού βασισμένο σε ροή, το οποίο αναπτύχθηκε από τις Υπηρεσίες Αναδυόμενης Τεχνολογίας της IBM και τώρα είναι μέρος του Ιδρύματος JS. Ο προγραμματισμός βασισμένος σε ροή ανακαλύφθηκε από τον J. Paul Morrison την δεκαετία του 1970 και είναι ένας τρόπος περιγραφής της συμπεριφοράς μιας εφαρμογής από ένα δίκτυο κόμβων. Ο κάθε κόμβος έχει έναν αριθμό εισόδων και εξόδων. Τα δεδομένα λαμβάνονται από τον κόμβο, εκτελεί κάποιο υπολογισμό και στη συνέχεια στέλνει τα δεδομένα στον επόμενο κόμβο. Το node-RED καθιστά τον προγραμματισμό πιο προσιτό στον χρήστη, καθώς προσφέρει την ευελιξία

στη δημιουργία εφαρμογών, κυρίως γύρω από τον τομέα του IoT, χωρίς να χρειάζεται ιδιαίτερη μέριμνα για ορισμένα κοινά εμπόδια προγραμματισμού. Κατά την ανάπτυξη προγράμματος στο node-RED, ο προγραμματιστής χρησιμοποιεί το web GUI για τη δημιουργία ροών ενώ παράλληλα προσφέρει την δυνατότητα προσαρμογής, επιτρέποντας στον χρήστη να δημιουργήσει νέους κόμβους χρησιμοποιώντας την Javascript. Η προεπιλεγμένη δομή δεδομένων που περνά από τον ένα κόμβο στον άλλο ονομάζεται "msg" ενώ συνήθως το κύριο στοιχείο εισόδου και εξόδου στους κόμβους είναι το χαρακτηριστικό "msg.payload". Ένα παράδειγμα από το γραφικό περιβάλλον μιας ροής στο node-RED φαίνεται στην Εικόνα 4.2-2. Μετά την ανάπτυξη μιας ροής, αυτή μπορεί να αποθηκευτεί σε αρχείο JSON. Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκε για την λήψη των δεδομένων από την εφαρμογή Torque, την επεξεργασία και την μεταφόρτωση τους στο Cloud.



Εικόνα 4.2-2: Γραφικό περιβάλλον σύνταξης ροών του Node-RED

4.2.4 Cloud Azure IoT

Το Azure είναι μια υπηρεσία Cloud Computing που αναπτύχθηκε από την Microsoft για την κατασκευή, την δοκιμή, την δημιουργία και την διαχείριση εφαρμογών και υπηρεσιών μέσω των κέντρων δεδομένων διαχειριζόμενα από την Microsoft. Παρέχει τις υπηρεσίες SaaS, PaaS και IaaS και υποστηρίζει πολλές διαφορετικές γλώσσες προγραμματισμού και εργαλεία συμπεριλαμβανομένων λογισμικού και συστημάτων της Microsoft αλλά και άλλων κατασκευαστών. Το Azure ανακοινώθηκε το 2008, ξεκίνησε με το κωδικό όνομα "Project Red Dog" και κυκλοφόρησε το 2010, ως Windows Azure πριν μετονομαστεί σε Microsoft Azure το 2014.

Το Azure προσφέρει έναν μεγάλο αριθμό υπηρεσιών, ορισμένες από τις οποίες αναφέρονται παρακάτω:

- Υπολογιστών
- Ταυτότητας
- Κινητών υπηρεσιών
- Διαχείρισης δεδομένων
- Αποθήκευσης
- Μηχανικής εκμάθησης
- Διαδικτύου των Πραγμάτων

Επίσης, παρέχει μια Διεπαφή Προγραμματισμού Εφαρμογών (API) που βασίζεται σε τεχνολογίες REST, HTTP και XML που επιτρέπουν στον προγραμματιστή να αλληλεπιδρά με τις υπηρεσίες που παρέχονται από το Microsoft Azure. Η Microsoft παρέχει επίσης μια βιβλιοθήκη κλάσης διαχειριζόμενη από πλευράς πελάτη, που ενσωματώνει τις λειτουργίες αλληλεπίδρασης με τις υπηρεσίες.

Το Azure IoT είναι μια συλλογή από υπηρεσίες Cloud που διαχειρίζεται η Microsoft και συνδέουν, παρακολουθούν και ελέγχουν δισεκατομμύρια συσκευές IoT. Στην Εικόνα 4.2-3 παρουσιάζεται η κεντρική σελίδα του Azure IoT Hub. Το IoT Hub αποτελεί μια υπηρεσία η οποία φιλοξενείται στο Cloud και λειτουργεί ως κεντρικός κόμβος μηνυμάτων για αμφίδρομη επικοινωνία μεταξύ της εφαρμογής IoT και των συσκευών που διαχειρίζεται. Υποστηρίζει την επικοινωνία από τη συσκευή στο Cloud και το αντίστροφο. Το IoT Hub υποστηρίζει πολλά μοτίβα ανταλλαγής μηνυμάτων, όπως την τηλεμετρία συσκευής προς Cloud, μεταφόρτωση αρχείων από συσκευές και μεθόδους απάντησης αιτήματος για τον έλεγχο των συσκευών από το Cloud. Τα πρωτόκολλα που υποστηρίζει για την σύνδεση των συσκευών είναι τα: HTTPS, AMQP και MQTT. Ακόμη, υπάρχει η δυνατότητα δημιουργίας συσκευής Edge μέσα στο IoT Hub η οποία συγκροτείται από τρία στοιχεία:

1. Τα IoT Edge modules, που είναι containers και εκτελούν υπηρεσίες του Azure, υπηρεσίες τρίτων ή τον κωδικό του χρήστη. Αναπτύσσονται σε συσκευές IoT Edge και εκτελούνται τοπικά σε αυτές τις συσκευές. Μια module αποτελείται από τέσσερα στοιχεία τα οποία αναφέρονται παρακάτω:

- α) Την module image, το πακέτο που περιέχει το λογισμικό που καθορίζει μια ενότητα.

β) Την *module instance*, είναι μια συγκεκριμένη υπολογιστική μονάδα που εκτελεί την *module image* σε μια συσκευή IoT Edge. Ξεκινάει από το IoT Edge runtime.

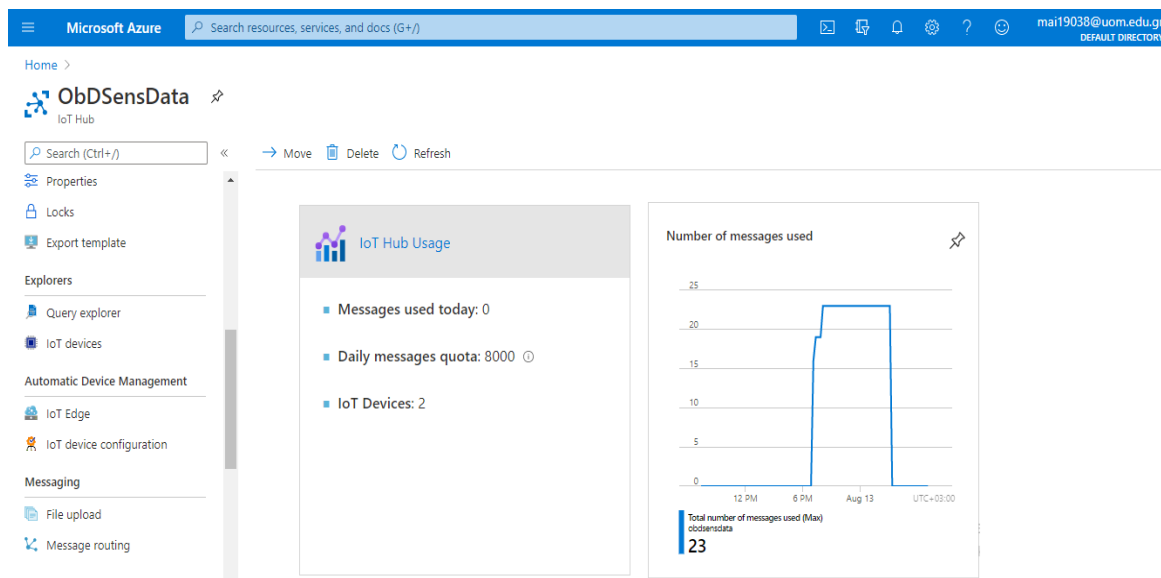
γ) Μια *module identity*, είναι μια πληροφορία (συμπεριλαμβανομένων των διαπιστευτηρίων ασφαλείας) αποθηκευμένη στο IoT Hub, η οποία σχετίζεται με κάθε *module instance*.

δ) Το *module twin*, είναι ένα έγγραφο JSON που είναι αποθηκευμένο στο IoT Hub, το οποίο περιέχει πληροφορίες κατάστασης μιας *module instance*, συμπεριλαμβανομένων μεταδεδομένων, διαμορφώσεων και συνθηκών.

2. Το IoT Edge runtime που εκτελείται σε κάθε συσκευή Edge και διαχειρίζεται τους *containers* που αναπτύσσονται σε κάθε συσκευή. Είναι μια συλλογή προγραμμάτων που μετατρέπουν μια συσκευή σε μια συσκευή IoT Edge. Συλλογικά, ο IoT Edge runtime επιτρέπει στις συσκευές IoT Edge να λαμβάνουν κώδικα για να τρέχουν στη συσκευή Edge και να κοινοποιούν τα αποτελέσματα. Ο runtime είναι υπεύθυνος για τις ακόλουθες λειτουργίες σε συσκευές IoT Edge: εγκατάσταση και ενημέρωση του φόρτου εργασίας στη συσκευή, διατήρηση των πρότυπων ασφαλείας Azure IoT Edge στη συσκευή, διασφαλίζει ότι οι IoT Edge modules εκτελούνται πάντα, Αναφέρει την κατάσταση της *module* στο Cloud για απομακρυσμένη παρακολούθηση, διαχειρίζεται την επικοινωνία μεταξύ των συσκευών, την επικοινωνία μεταξύ των *module* στη συσκευή IoT Edge και την επικοινωνία μεταξύ της συσκευής IoT Edge και του Cloud.

3. Μια διεπαφή βασισμένη σε Cloud που επιτρέπει την εξ αποστάσεως παρακολούθηση και διαχείριση συσκευών Edge.

Στην εργασία χρησιμοποιήθηκε η υπηρεσία Azure IoT Hub όπου δημιουργήθηκαν, μια συσκευή IoT και μια συσκευή Edge IoT για την μεταφόρτωση μηνυμάτων από την εφαρμογή Torque και την μετέπειτα σύγκριση των αποτελεσμάτων σε επίπεδο καθυστέρησης και εύρους ζώνης.



Εικόνα 4.2-3: Γραφικό περιβάλλον Azure IoT Hub

4.2.5 Docker

Το Docker είναι μια πλατφόρμα λογισμικού ανοιχτού κώδικα που υλοποιεί Εικονικοποίηση (Virtualization) σε επίπεδο Λειτουργικού Συστήματος. Ουσιαστικά πρόκειται για μια μηχανή κοντέινερ (container) που χρησιμοποιεί τις δυνατότητες του πυρήνα Linux, όπως χώρους ονομάτων και ομάδες ελέγχου, για τη δημιουργία container πάνω από ένα λειτουργικό σύστημα. Ένα container είναι μια τυπική μονάδα λογισμικού που περιέχει τον κώδικα και όλες τις εξαρτήσεις του, έτσι ώστε η εφαρμογή να εκτελείται γρήγορα και αξιόπιστα από το ένα υπολογιστικό περιβάλλον στο άλλο. (“What is a Container”, 2020). Ορισμένα πλεονεκτήματα του Docker για καταναεμημένες εφαρμογές είναι η μειωμένη επιβάρυνση του επεξεργαστή, η φορητότητα και η βελτιωμένη απόδοση δικτύου (Felter et al. (2015)). Επίσης είναι διαθέσιμο για λειτουργικά συστήματα Linux και Windows. Όπως αναφέρθηκε στην προηγούμενη υποπαράγραφο, οι IoT Edge modules είναι containers και ως μονάδες εκτέλεσης χρειάζονται μια μηχανή Docker για να τις εκτελέσει. Για την εξυπηρέτηση αυτού του σκοπού, εγκαταστάθηκε στο Raspberry Pi η Docker engine.

4.2.6 Πρωτόκολλο HTTP

Το πρωτόκολλο μεταφοράς υπερκειμένου (Hypertext Transfer Protocol) είναι ένα πρωτόκολλο του επιπέδου εφαρμογής για καταναεμημένα, συνεργατικά συστήματα πληροφοριών υπερμέσων και είναι το θεμέλιο της επικοινωνίας δεδομένων για τον Παγκόσμιο Ιστό (Fielding et al. (1999)).

Το HTTP λειτουργεί ως πρωτόκολλο αίτησης-απόκρισης στο υπολογιστικό μοντέλο πελάτη-διακομιστή. Αυτό σημαίνει ότι πελάτης μπορεί να είναι ένα πρόγραμμα περιήγησης ιστού (browser) και διακομιστής μια εφαρμογή που εκτελείται σε έναν υπολογιστή. Ο πελάτης στέλνει ένα μήνυμα αίτησης HTTP στον διακομιστή και στην συνέχεια ο δεύτερος επιστρέφει ένα μήνυμα απόκρισης στον πελάτη. Επομένως, το HTTP είναι ένα πρωτόκολλο επικοινωνίας που βασίζεται σε TCP / IP, το οποίο χρησιμοποιείται για την παράδοση δεδομένων (αρχεία HTML, αρχεία εικόνας, αποτελέσματα ερωτημάτων κ.λ.π.) στον Παγκόσμιο Ιστό. Η προεπιλεγμένη θύρα είναι η TCP 80, αλλά μπορούν να χρησιμοποιηθούν και άλλες θύρες. Παρέχει έναν τυποποιημένο τρόπο επικοινωνίας των υπολογιστών μεταξύ τους. Η προδιαγραφή του HTTP καθορίζει με ποιο τρόπο τα δεδομένα αιτημάτων των πελατών θα συνταχθούν και θα αποσταλούν στο διακομιστή καθώς και πώς οι διακομιστές ανταποκρίνονται στα αιτήματα αυτά. Όσον αφορά το επίπεδο μετάδοσης, το HTTP χρησιμοποιεί συνήθως το TCP μπορεί όμως να προσαρμοστεί ώστε να χρησιμοποιεί και UDP.

Το HTTP όπως αναφέρθηκε προηγουμένως στέλνει σε κάποιον διακομιστή του διαδικτύου, για την εγκαθίδρυση της επικοινωνίας, ένα μήνυμα αίτησης μέσω κάποιου προγράμματος περιήγησης ιστού που βρίσκεται στον Η/Υ του χρήστη, για την αίτηση κάποιας ενέργειας. Οι ενέργειες αυτές, γνωστές και ως μέθοδοι, αναφέρονται στον ακόλουθο πίνακα (Πίνακας 4.2-1).

Πίνακας 4.2-1: Περιγραφή μεθόδων HTTP (Fielding et al., 1999)

ΜΕΘΟΔΟΣ	ΠΕΡΙΓΡΑΦΗ ΜΕΘΟΔΟΥ
GET	Χρησιμοποιείται για την ανάκτηση πληροφοριών από τον δεδομένο διακομιστή.
POST	Χρησιμοποιείται για την αποστολή δεδομένων στον διακομιστή, όπως, πληροφορίες, μεταφόρτωση αρχείων κ.λ.π. χρησιμοποιώντας φόρμες HTML
HEAD	Ίδια με την GET, αλλά μεταφέρει μόνο τη γραμμή κατάστασης και την ενότητα κεφαλίδας.
PUT	Αιτείται την αποθήκευση ενός επισυναπτόμενου τμήματος κάτω από το παρεχόμενο URI

OPTIONS	Επιστρέφει τις μεθόδους HTTP που υποστηρίζει ο διακομιστής για την καθορισμένη διεύθυνση URL
DELETE	Αίτημα που διαγράφει τον καθορισμένο πόρο
CONNECT	Μετατρέπει τη σύνδεση αιτήματος σε μια ασφαλή σήραγγα TCP / IP
TRACE	Αντικατοπτρίζει το ληφθέν αίτημα, έτσι ώστε ο πελάτης να εντοπίζει τυχόν αλλαγές ή προσθήκες από ενδιάμεσους διακομιστές

Στην παρούσα εργασία χρησιμοποιείται για την επικοινωνία της εφαρμογής Torque με το node-RED και για την αποστολή δεδομένων στο Cloud.

4.3 Περιβάλλον πειράματος

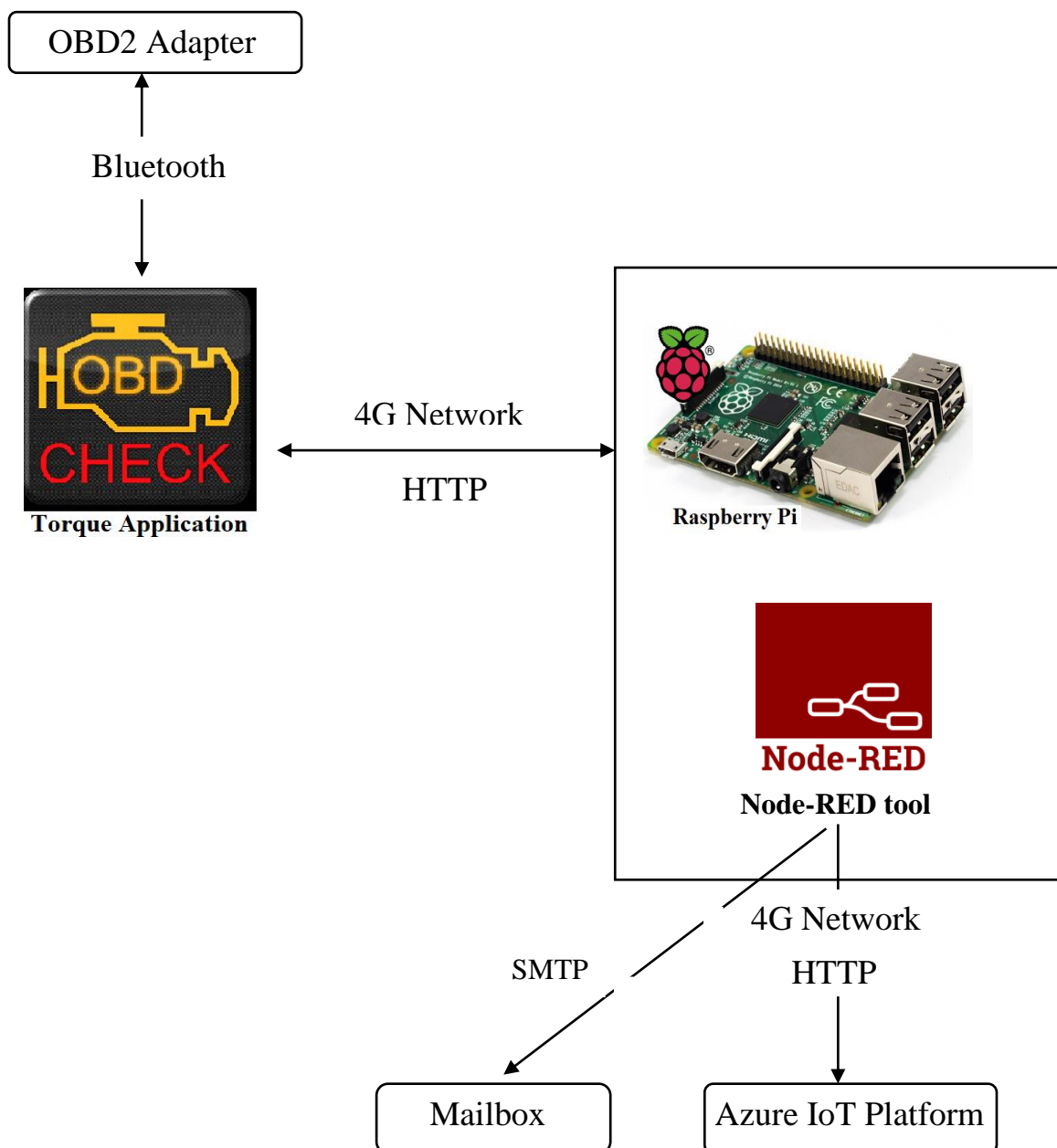
Στον παρακάτω πίνακα (Πίνακας 4.3-1) παρουσιάζονται συγκεντρωτικά κάποια τεχνικά χαρακτηριστικά των hardware και software που χρησιμοποιήθηκαν για την υλοποίηση του πειράματος.

Πίνακας 4.3-1: Τεχνικά χαρακτηριστικά και εκδόσεις εργαλείων πειράματος

Lenovo B51	i3-6006U, 2core, 4GB RAM	Windows 10 OS
Huawei P9	ARM Cortex-A53, 8score, 3GB RAM	Android 6.0
Raspberry Pi 4	ARMv8, 4core, 2GB RAM	Raspbian
Nissan NV300	ISO 15765 - 4CAN 11/500	
OBDII Protocol		
Docker	v. 19.03.8	
Wireshark	v. 3.2.3	
Node-RED	v. 1.0.6	
Torque application	v. 1.10.114	
Microsoft Azure	IoT Hub Service	
ELM327	v. 1.5	

4.4 Υλοποίηση πειραματικής προσέγγισης

Στο υποκεφάλαιο αυτό παρουσιάζονται λεπτομερέστερα τα βήματα για την σύνθεση του πειραματικού περιβάλλοντος και αναλύονται οι λειτουργίες του κάθε υποσυστήματος. Αρχικά αξίζει να αναφερθεί ότι και στα δύο πειράματα χρησιμοποιήθηκε το ίδιο όχημα Nissan NV300, το διαγνωστικό ELM327, η εφαρμογή Torque, το εργαλείο node-RED (εγκαταστάθηκε στην φορητή υπολογιστική μονάδα και το Raspberry) και ο απομακρυσμένος server της Azure. Για την υλοποίηση του πρώτου πειράματος, στην θύρα OBDII του οχήματος συνδέθηκε το ELM327 και μέσω Bluetooth επικοινωνεί αυτό με την εφαρμογή Torque. Το αυτοκίνητο μετακινείται εντός ελεγχόμενου χώρου όπου η παροχή δικτύου είναι 4G και μέσω πολλαπλών Access Points, το smartphone που φέρει την εφαρμογή Torque και βρίσκεται στο αυτοκίνητο, παραμένει συνδεδεμένο στο 4G. Επίσης, εντός του ελεγχόμενου χώρου, σε κάποια απόσταση από το αυτοκίνητο η οποία είναι μεταβλητή λόγω της μετακίνησής του, υπάρχει μια φορητή υπολογιστική μονάδα συνδεδεμένη στο ίδιο δίκτυο 4G, στην οποία έχει εγκατασταθεί το node-RED για την παραλαβή και προώθηση των δεδομένων. Κατά την υλοποίηση του δεύτερου πειράματος, ανατίθεται το Raspberry Pi ως συσκευή Edge επί του οχήματος, έτσι ώστε τα δεδομένα που λαμβάνονται από την θύρα OBDII να επεξεργάζονται στην συσκευή Edge και στη συνέχεια να αποστέλλονται στο Core Cloud. Στην περίπτωση αυτή εφόσον η υπολογιστική συσκευή Edge βρίσκεται εγκατεστημένη στο αυτοκίνητο, το δίκτυο 4G παρέχεται από το smartphone που λειτουργεί ως Access Point. Επίσης, στο δεύτερο πείραμα έχει γίνει προσαρμογή στη συσκευή Edge ώστε να φιλτράρει τα δεδομένα και να στέλνει ειδοποίηση στο e-mail του χρήστη όταν οι τιμές των αισθητήρων δεν είναι φυσιολογικές. Παρακάτω παρουσιάζεται το διάγραμμα αρχιτεκτονικής του συστήματος Edge (Διάγραμμα 4-1) όπου φαίνονται τα στοιχεία που χρησιμοποιήθηκαν και στη συνέχεια ακολουθεί η ανάλυση της προετοιμασίας των εργαλείων.



Διάγραμμα 4-1: Αρχιτεκτονική συστήματος Edge

4.4.1 Εξυπηρετητής Azure – Πελάτης Raspberry

Αρχικά, δημιουργήθηκε στην πλατφόρμα Azure ένα IoT Hub, με την ονομασία “ObDSensData” όπου θα μεταφορτώνονται όλα τα δεδομένα. Το IoT Hub είναι μια διαχειριζόμενη υπηρεσία που εμπεριέχεται στο Cloud και λειτουργεί ως κεντρικός κόμβος μηνυμάτων για αμφίδρομη επικοινωνία μεταξύ της εφαρμογής IoT και των συσκευών που διαχειρίζεται. Το IoT Hub υποστηρίζει την επικοινωνία από τη συσκευή στο σύννεφο και αντίστροφα, τόσο για ανταλλαγή μηνυμάτων όσο και για μεταφόρτωση

αρχείων. Στην συνέχεια έγινε προσθήκη στην υπηρεσία μιας συσκευής IoT για το 1^ο πείραμα και μιας συσκευής Edge IoT για το 2^ο πείραμα στην οποία ανατέθηκε το Edge runtime. Το runtime αποτελείται από τα στοιχεία:

1) security daemon, ξεκινάει με την έναρξη λειτουργίας της συσκευής και παράλληλα εκκινεί τον IoT Edge agent.

2) IoT Edge agent, διευκολύνει την ανάπτυξη και παρακολούθηση των modules στη συσκευή IoT Edge, συμπεριλαμβανομένου του IoT Edge Hub.

3) IoT Edge hub, διαχειρίζεται την επικοινωνία μεταξύ των modules στη συσκευή IoT Edge και μεταξύ της συσκευής και του IoT Hub.

Κατά την δημιουργία του runtime παρέχεται μια συμβολοσειρά σύνδεσης συσκευής (connection string).

Στη συνέχεια, στο Raspberry δόθηκαν εντολές για να λειτουργήσει ως συσκευή Edge, μερικές από τις οποίες παρατίθενται παρακάτω:

Εγκατάσταση διαμόρφωσης αποθετηρίου:

```
Curl https://packages.microsoft.com/config/debian/stretch/multiarch/prod.list >
./microsoft-prod.list
```

Εγκατάσταση δημοσίου κλειδιού GPG Microsoft:

```
curl https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > microsoft.gpg
sudo cp ./microsoft.gpg /etc/apt/trusted.gpg.d/
```

Εγκατάσταση Docker-ce:

```
curl -sSL get.docker.com | sh && sudo usermod pi -aG docker
```

Εγκατάσταση IoT Edge security daemon:

```
sudo apt-get install iotedge
```

Στη συνέχεια έγινε διαμόρφωση του security daemon έτσι ώστε να συνδεθεί η φυσική συσκευή (Raspberry) με μια ταυτοποιημένη συσκευή (EdgeOBD2) που δημιουργήθηκε στο IoT Hub του Azure.

Για την μεταφόρτωση μηνυμάτων στο IoT Hub, εγκαταστάθηκε στο Raspberry το container node-RED το οποίο βρίσκεται στην ηλεκτρονική διεύθυνση <https://hub.docker.com/u/gbbiotwesouth>.

Ακολούθως πραγματοποιήθηκε η εισαγωγή του συγκεκριμένου module στο runtime της συσκευής EdgeOBD2 μέσα από την επιλογή Add→IoT Edge module, όπως φαίνεται στην Εικόνα 4.4-1.

Microsoft Azure Search resources, services, and docs (G+/)

Home > ObDSensData > EdgeOBD2 >

Set modules on device: EdgeOBD2

ObDSensData

Modules Routes Review + create

IoT Edge Modules

An IoT Edge module is a Docker container you can deploy to IoT Edge devices. It communicates with other modules and sends data to the IoT Edge runtime. Using this UI you can import Azure Service IoT Edge modules or specify the settings for an IoT Edge module. Setting modules on each device will be counted towards the quota and throttled based on the IoT Hub tier and units. For example, for S1 tier, modules can be set 10 times per second if no other updates are happening in the IoT Hub. [Learn more](#)

+ Add Runtime Settings

NAME	DESIRED STATUS
nodered	running

Review + create < Previous Next: Routes >

Εικόνα 4.4-1: Εισαγωγή container στην συσκευή Edge του Azure

Για να ολοκληρωθεί η διαδικασία εγκατάστασης module προστέθηκε και η διαδρομή των μηνυμάτων κατά την μεταφόρτωση. Στην Εικόνα 4.4-2 στην καρτέλα Routes συμπληρώθηκε στην θέση VALUE: FROM /messages/modules/{noderedmodule}/outputs/* INTO \$upstream, που σημαίνει ότι η διαδρομή στέλνει τα μηνύματα του node-RED module στο IoT Hub.

Microsoft Azure Search resources, services, and docs (G+/)

Home > EdgeOBD2 >

Set modules on device: EdgeOBD2

ObDSensData

Modules Routes Review + create

Routes

You can set routes between modules, which gives you the flexibility to send messages where they need to go without the need for additional services to process messages or to write additional code. [Learn more](#)

NAME	VALUE
routeToHub	FROM /messages/modules/(noderedmodule)/outputs/* INTO \$upstream
Route name	FROM /messages/* INTO \$upstream

Review + create < Previous Next: Review + create >

Εικόνα 4.4-2: Εισαγωγή διαδρομής μηνυμάτων

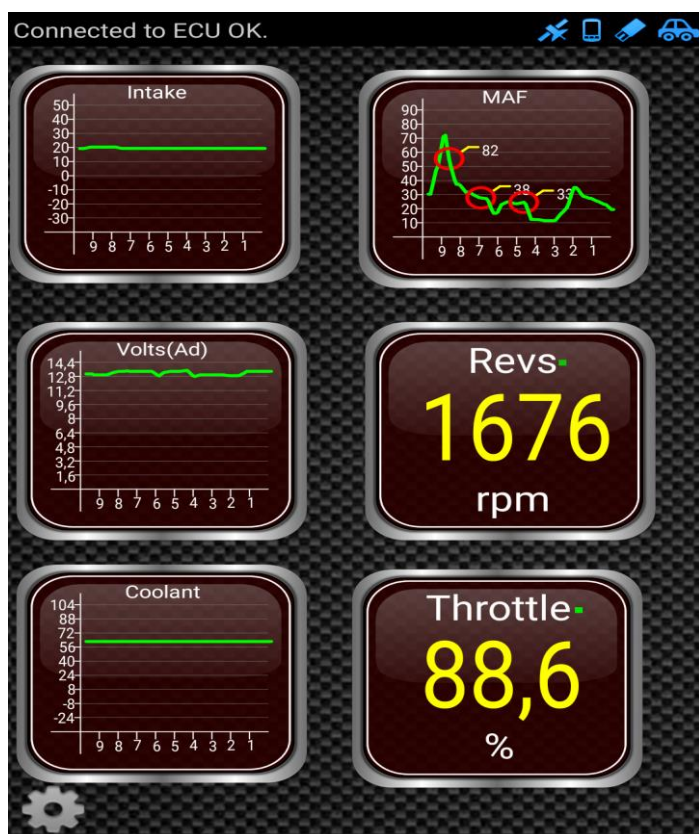
4.4.2 Σύνδεση OBDII reader με την εφαρμογή Torque

Η σύνδεση του ELM327 με την εφαρμογή Torque είναι απαραίτητη πρωτίστως για την μετατροπή των εισερχόμενων δεδομένων από την θύρα OBDII σε μετρήσιμες τιμές αισθητήρων και δευτερευόντως για την αποστολή των τιμών στον web server.



Εικόνα 4.4-3: Επικοινωνία ELM327 με Torque

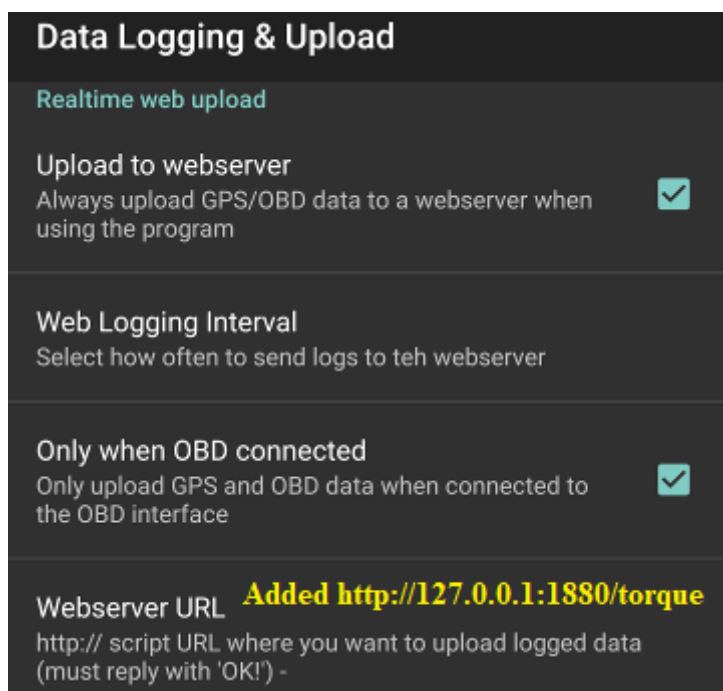
Στην Εικόνα 4.4-3 έχει τοποθετηθεί το ELM327 στην θύρα του NISSAN NV300. Για την ενεργοποίηση του ELM, ενεργοποιήθηκε αρχικά το Bluetooth στο smartphone και, μετά από κάποιο χρόνο λειτουργίας του κινητήρα μετρήθηκαν οι τιμές των αισθητήρων: θερμοκρασίας εισερχόμενου αέρα, μάζας αέρα, θερμοκρασίας ψυκτικού υγρού, στροφών, θέσης επιταχυντή και τάσης εξόδου όπως απεικονίζονται στην Εικόνα 4.4-4 και στα δυο πειράματα.



Εικόνα 4.4-4: Πραγματικές τιμές αισθητήρων

4.4.3 Μεταφόρτωση δεδομένων μέσα από την εφαρμογή Torque

Η εφαρμογή Torque παρέχει την δυνατότητα αποστολής των καταγεγραμμένων δεδομένων πραγματικού χρόνου σε κάποιο web server. Μπορεί να οριστεί ο χρόνος καταγραφής των δεδομένων από την επιλογή *File logging interval* των ρυθμίσεων *Data logging and Upload* (Εικόνα 4.4-5). Επίσης έγινε επιλογή του *Upload to web server* και στο πεδίο *web server URL* τοποθετείται το URL του server που επιθυμεί ο χρήστης να φορτώσει τα δεδομένα. Το node-red δημιουργεί μόνο του και χρησιμοποιεί τον ExpressJS server, ο οποίος παρέχει γενικά για την χρήση web server, URL του τύπου <http://{your-server-ip}:1880>. Εφόσον και στα δυο πειράματα το smartphone και η υπολογιστική μονάδα είναι συνδεδεμένα στο ίδιο δίκτυο, καθίσταται δυνατή η ροή των δεδομένων από την εφαρμογή Torque στο localhost της υπολογιστικής μονάδας. Ο τρόπος με τον οποίο λαμβάνονται τα δεδομένα στον web server από την μεριά του node-RED, καθώς και η ροή που αναπτύχθηκε, περιγράφονται στο επόμενο υποκεφάλαιο.

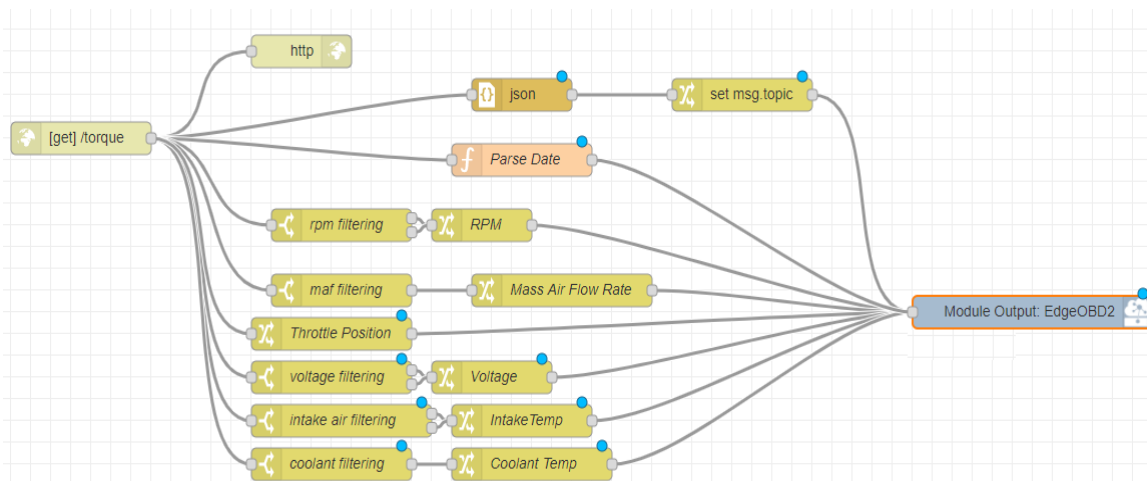


Εικόνα 4.4-5: Ρυθμίσεις και επιλογές καταγραφής δεδομένων

4.4.4 Ανάπτυξη ροής στο Node-RED και ανάλυση των κόμβων

Η ροή που αναπτύχθηκε περιλαμβάνει τους κόμβους http in, http out, function, switch, change, email, Azure IoT Hub και module output της βιβλιοθήκης Azure IoT Edge.

Στην Εικόνα 4.4-6 παρουσιάζεται η ροή που χρησιμοποιήθηκε στο 2^ο πείραμα και παρακάτω αναλύονται οι κόμβοι. Στο 1^ο πείραμα η ροή παραμένει η ίδια με την διαφορά ότι δεν φιλτράρονται τα εισερχόμενα δεδομένα. Η χρήση της φορητής υπολογιστικής μονάδας και του node-RED γίνεται μόνο για την σύγκριση των αποτελεσμάτων με τα αποτελέσματα του 2^{ου} πειράματος διότι σε συνθήκες κανονικής οδήγησης σε οποιοδήποτε οδικό δίκτυο δεν παρέχεται από τις υποδομές κάποιος server κοντά στο μετακινούμενο όχημα για την επεξεργασία των μετρήσεων του. Για τον λόγο αυτό, αφαιρείται η διαδικασία φιλτραρίσματος από την ροή node-RED της φορητής υπολογιστικής μονάδας και συνεπώς οι κόμβοι switch.



Εικόνα 4.4-6: Ροή Node-RED της προτεινόμενης Edge αρχιτεκτονικής

Network Nodes

Η συλλογή των δεδομένων ξεκινάει με χρήση των κόμβων http-in/out που όταν χρησιμοποιούνται ως ζεύγος επιτρέπουν την λήψη εισερχόμενων αιτημάτων ιστού και την απάντηση σε αυτά. Στην προκειμένη περίπτωση ο κόμβος http in αιτείται και διαχωρίζει τα δεδομένα, ενώ ο http out δίνει την απάντηση “OK!”, την οποία περιμένει η εφαρμογή Torque. Στην ρύθμιση του κόμβου http in απλώς προστέθηκε η μέθοδος GET και στο πεδίο URL το path /torque. Ο κόμβος http out ρυθμίστηκε με Header το “OK!”.

Function Nodes

Ακολούθως χρησιμοποιήθηκαν οι κόμβοι json και change για την προετοιμασία και τον ορισμό των ιδιοτήτων του μηνύματος που απαιτούνται μετά το http get και ο ορισμός είναι από msg.topic σε torque\raw. Εάν μια ιδιότητα μηνύματος περιέχει μια συμβολοσειρά JSON, πρέπει πρώτα να αναλυθεί στο αντίστοιχο αντικείμενο JavaScript πριν να είναι δυνατή η πρόσβαση στις ιδιότητες που περιέχει. Επίσης έγινε χρήση του κόμβου function (δέχεται εγγραφή συναρτήσεων σε Javascript) για την εγγραφή ημερομηνίας : `msg.payload.time = new Date().getTime()` και για την προετοιμασία αποστολής email σε περίπτωση σφάλματος.

Ο κόμβος change αξιοποιήθηκε επίσης για την μεταφορά του msg.payload.xxx του κάθε αισθητήρα στο κεντρικό msg.payload και ακολούθως για να οριστεί το msg.topic στο string: torque/vehicle/xxx (στην θέση xxx εισάγεται η ονομασία του κόμβου του αισθητήρα). Η διαδικασία αυτή έγινε για όλους τους αισθητήρες που μετρώνται οι τιμές τους. Στο 2^ο πείραμα, μπροστά από τον κόμβο change τοποθετείται ο κόμβος switch για κάθε αισθητήρα (εκτός του επιταχυντή) ο οποίος πραγματοποιεί

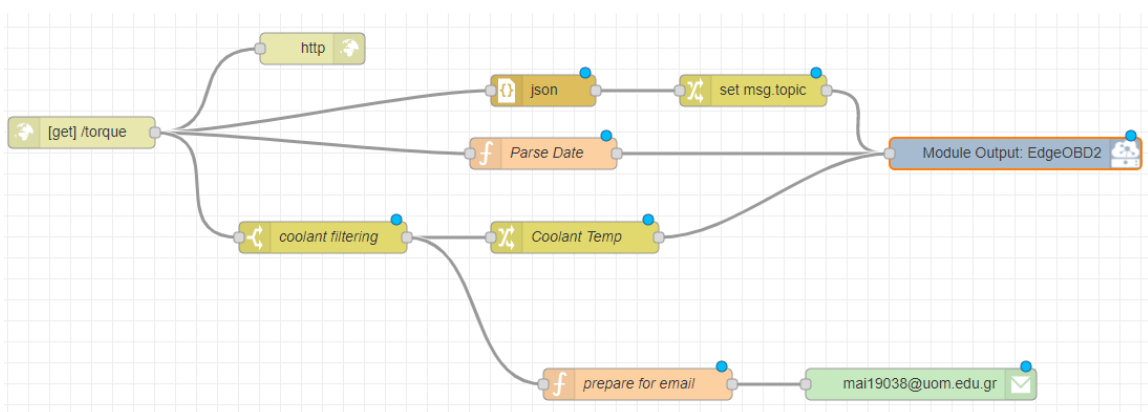
filtering ούτως ώστε να μην αποστέλλονται όλα τα δεδομένα στο Hub αλλά μόνο τα κρίσιμα. Για την περίπτωση όπου αποστέλλεται ειδοποιητικό email στον χρήστη σε περίπτωση σφάλματος χρησιμοποιείται επίσης ο κόμβος switch για τον έλεγχο των αναμενόμενων τιμών. Επί παραδείγματι, στην Εικόνα 4.4-7 αναπαρίσταται η ίδια ροή, ενδεικτικά για έναν αισθητήρα μόνο, αυτόν της μέτρησης θερμοκρασίας ψυκτικού υγρού, με την προσθήκη της λειτουργίας άμεσης ειδοποίησης με email. Ο κόμβος switch coolant filtering ελέγχει αν οι εισερχόμενες τιμές στον κόμβο είναι φυσιολογικές και σε περίπτωση που εντοπίσει μη φυσιολογική τιμή πραγματοποιείται αποστολή email.

Email Node

Ο κόμβος email εξόδου είναι υπεύθυνος για την αποστολή του μηνύματος στο λογαριασμό email που δηλώνει ο χρήστης με χρήση πρωτοκόλλου SMTP στην πόρτα 465.

Azure IoT Nodes

Στο 2^ο πείραμα ως τελικός κόμβος που φορτώνει τα δεδομένα στο IoT Hub είναι ο Azure IoT Edge Module Output. Επιτρέπει την αποστολή μηνυμάτων με προϋπόθεση το node-red να τρέχει σαν module στην συσκευή Edge που δημιουργήθηκε, εισάγοντας στο πεδίο output το όνομα της edge συσκευής -EdgeOBD2- στο Cloud Azure IoT Hub. Η αποστολή γίνεται με χρήση HTTP. Επίσης, για να γίνει συγκριτική ανάλυση των πειραματικών αποτελεσμάτων χρησιμοποιήθηκε στην ροή του 1^{ου} πειράματος ο κόμβος Azure IoT Hub ως τελικός κόμβος και πάλι. Η αποστολή των μηνυμάτων γίνεται και εδώ με την χρήση HTTP και του hostname της συσκευής IoT του Azure Cloud.



Εικόνα 4.4-7: Ροή node-RED με λειτουργία άμεσης ειδοποίησης σφάλματος ενός αισθητήρα

4.5 Παρακολούθηση απόδοσης δικτύου

Για την μέτρηση των καθυστερήσεων και του εύρους ζώνης χρησιμοποιήθηκε και στα δυο πειράματα το πρόγραμμα Wireshark. Το Wireshark συλλέγει και αναλύει πακέτα κατά την επικοινωνία και αποστολή δεδομένων. Επιλέχθηκε το συγκεκριμένο εργαλείο διότι παρέχει διάφορες επιλογές απεικόνισης γράφων, είναι εφικτή η εφαρμογή φίλτρων για τον περιορισμό του εύρους των δεδομένων που συλλέγει και είναι εύκολο στην χρήση. Στην προκειμένη περίπτωση αξιοποιήθηκε για την συλλογή πακέτων κατά την αποστολή τους από το node-RED στην πλατφόρμα Azure IoT Hub. Έτσι εγκαταστάθηκε και στις δυο υπολογιστικές μονάδες. Οι λειτουργίες που χρησιμοποιήθηκαν είναι της απεικόνισης γράφου Round Trip Time και Input/Output. Το Round Trip Time δηλώνει την χρονική διάρκεια που απαιτείται για την μετάδοση ενός σήματος από τον αποστολέα στον παραλήπτη, συν την χρονική διάρκεια που απαιτείται για να σταλεί πίσω στον αποστολέα η επιβεβαίωση της λήψης του σήματος από τον παραλήπτη. Έτσι προσδιορίζεται η καθυστέρηση μετάδοσης. Το RTT μετράται σε ms. Το εύρος ζώνης μετράται ως το ποσό των δεδομένων που μπορούν να μεταφερθούν από ένα σημείο σε άλλο εντός ενός δικτύου σε ένα συγκεκριμένο χρονικό διάστημα. Συνήθως, το εύρος ζώνης εκφράζεται ως ρυθμός bit και μετράται σε bit ανά δευτερόλεπτο. Το γράφημα I/O εμφανίζει τη συνολική κίνηση των πακέτων εντός του δικτύου και μετράει σε bit ή packets / sec. Μέσα από το συγκεκριμένο γράφημα προσδιορίζεται η κατανάλωση εύρους ζώνης στα πειράματα.

5 Πειραματική ανάλυση

Στόχος της διπλωματικής εργασίας είναι η μελέτη της απόδοσης ενός Edge Cloud έναντι του απομακρυσμένου Core Cloud, κατά την μετάδοση δεδομένων από την θύρα OBDII του αυτοκινήτου. Πιο συγκεκριμένα, αυτό μπορεί να επιτευχθεί μέσα από την παρακολούθηση της καθυστέρησης και του εύρους ζώνης σε δυο πειράματα με διαφορετική αρχιτεκτονική περιβάλλοντος.

Στόχος του 1^{ου} πειράματος είναι η διεξαγωγή μιας γενικής εικόνας των τιμών μέτρησης της καθυστέρησης και του εύρους ζώνης όταν το αυτοκίνητο χρησιμοποιεί, για τη μεταφόρτωση δεδομένων στο Core Cloud, δίκτυο 4G που εκπέμπεται σε μεγάλη εμβέλεια εντός ελεγχόμενου χώρου. Τα αποτελέσματα θα αξιοποιηθούν για την διενέργεια συγκριτικής ανάλυσης.

Στο 2^ο πείραμα στόχος είναι να ελεγχθεί εάν η προσθήκη συσκευής Edge στο αυτοκίνητο βελτιώνει αρχικά τις τιμές μέτρησης καθυστέρησης και εύρους ζώνης και σε δεύτερο στάδιο πόσο τοις εκατό τις βελτιώνει έτσι ώστε να αναδειχθούν οι ωφέλειες του Edge Computing σε πραγματικές συνθήκες.

5.1 Πειραματική μεθοδολογία

Πείραμα 1^ο

Αρχικά, ενεργοποιείται ο node-RED server στη φορητή υπολογιστική μονάδα που «τρέχει» στην διεύθυνση IP 127.0.0.1:1880 και στη συνέχεια εκτελείται από τον browser, στο URL <http://localhost:8080>, η ροή που δημιουργήθηκε στο node-RED. Έπειτα, εντός του οχήματος συνδέεται το ELM327 με την εφαρμογή Torque στο smartphone. Στην εφαρμογή Torque, από την καρτέλα *File logging interval* ρυθμίζεται η καθυστέρηση καταγραφής σε 5 sec. Αυτό γίνεται διότι στο Cloud Azure IoT Hub φορτώνονται όλα τα καταγεγραμμένα δεδομένα των 6 αισθητήρων χωρίς επεξεργασία. Επίσης στην καρτέλα *Webserver URL* της Torque εισάγεται το URL: <http://127.0.0.1:1880/torque>. Ακολουθεί η έναρξη της συλλογής πακέτων από το Wireshark. Με την εκκίνηση του κινητήρα, τα δεδομένα αποστέλλονται στο Azure IoT Hub. Τα μηνύματα των μετρήσεων που φορτώνονται στο Core Cloud από τους αισθητήρες μέσω της φορητής υπολογιστικής μονάδας φαίνονται στην Εικόνα 5.1-1.

```

Protocol: http
Connection string :HostName=ObdSensData.azure-devices.net;DeviceId=IoT0BDII;SharedAccessKey=-
05 Aug 14:09:50 - [info] [azureiothub:Azure IoT Hub] Connected to Azure IoT Hub.
05 Aug 14:09:50 - [info] [azureiothub:Azure IoT Hub] Message is pending. Sending it to Azure IoT Hub.
05 Aug 14:09:50 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":11,"coolantTemperature":85,"maf":20,"throttlePosition":20,"intakeAirTemp":23,"ts":1598699390
743}}
05 Aug 14:09:51 - [info] [azureiothub:Azure IoT Hub] Message sent.
05 Aug 14:09:55 - [info] [azureiothub:Azure IoT Hub] JSON
05 Aug 14:09:55 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":10,"coolantTemperature":93,"maf":65,"throttlePosition":11,"intakeAirTemp":23,"ts":1598699395
743}}
05 Aug 14:09:56 - [info] [azureiothub:Azure IoT Hub] Message sent.
05 Aug 14:10:00 - [info] [azureiothub:Azure IoT Hub] JSON
05 Aug 14:10:03 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":12,"coolantTemperature":95,"maf":51,"throttlePosition":28,"intakeAirTemp":23,"ts":1598699400
744}}
05 Aug 14:10:04 - [info] [azureiothub:Azure IoT Hub] Message sent.
05 Aug 14:10:05 - [info] [azureiothub:Azure IoT Hub] JSON
05 Aug 14:10:05 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":11,"coolantTemperature":87,"maf":64,"throttlePosition":54,"intakeAirTemp":24,"ts":1598699405
743}}
05 Aug 14:10:10 - [info] [azureiothub:Azure IoT Hub] JSON
05 Aug 14:10:10 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":13,"coolantTemperature":90,"maf":25,"throttlePosition":36,"intakeAirTemp":24,"ts":1598699410
744}}
05 Aug 14:10:11 - [info] [azureiothub:Azure IoT Hub] Message sent.
05 Aug 14:10:15 - [info] [azureiothub:Azure IoT Hub] JSON
05 Aug 14:10:15 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":11,"coolantTemperature":81,"maf":26,"throttlePosition":25,"intakeAirTemp":24,"ts":1598699415
748}}
05 Aug 14:10:16 - [info] [azureiothub:Azure IoT Hub] Message sent.
05 Aug 14:10:20 - [info] [azureiothub:Azure IoT Hub] JSON
05 Aug 14:10:20 - [info] [azureiothub:Azure IoT Hub] Sending Message to Azure IoT Hub :
  Payload: {"d":{"voltage":13,"coolantTemperature":95,"maf":54,"throttlePosition":14,"intakeAirTemp":23,"ts":1598699420
745}}

```

Εικόνα 5.1-1: Μηνύματα τιμών αισθητήρων από φορητή υπολογιστική μονάδα προς το Azure IoT Hub.

Πείραμα 2^ο

Εδώ, ενεργοποιείται το Raspberry Pi και ακολούθως τοποθετώντας στο πεδίο URL του browser την διεύθυνση IP που του ανατέθηκε (192.168.43.204:1880) από το Access Point στην θύρα 1880, εκτελείται η ροή που δημιουργήθηκε για την συσκευή Edge. Στη συνέχεια, συνδέεται το ELM327 με την εφαρμογή Torque, όπως και στο πείραμα 1. Σε αυτό το πείραμα, η καθυστέρηση καταγραφής *File logging interval* ρυθμίζεται στο 1 sec. Επειδή στο Cloud αποστέλλονται μόνο τα κρίσιμα δεδομένα των 6 αισθητήρων και έτσι αναμένεται μικρότερο bandwidth, η εφαρμογή ρυθμίστηκε να καταγράφει-φορτώνει δεδομένα κάθε 1 sec, με σκοπό την παρακολούθηση της καθυστέρησης και του bandwidth όταν επιβαρύνεται το δίκτυο. Επιπρόσθετα, η ροή node-RED για τη συσκευή Edge περιλαμβάνει τη λειτουργία αποστολής ειδοποιητικού e-mail στην περίπτωση όπου ανιχνευτεί μη φυσιολογική τιμή σε κάποιο αισθητήρα. Ρυθμίστηκαν 2 αισθητήρες τεχνητά ώστε να στέλνουν ειδοποίηση βλάβης κάθε 1 sec με σκοπό την επιμέρους επιβάρυνση του δικτύου, ώστε να παρατηρηθεί το αντίκτυπο στο εύρος ζώνης. Στην καρτέλα *Webserver URL* της Torque εισάγεται το URL: <http://192.168.43.204:1880/torque>. Ακολουθεί η έναρξη του Wireshark για μέτρηση RTT και bandwidth. Με την εκκίνηση του κινητήρα, ξεκινάει η ροή δεδομένων στο Azure IoT

Hub. Στην Εικόνα 5.1-2 παρουσιάζονται τα αντίστοιχα μηνύματα που στέλνονται από τη συσκευή Edge προς το Azure IoT Hub με την χρήση container.

```
12 May 13:33:07 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:08 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "12.124956655189173 V", "Temperature": "88.42659462981366 oC", "Engine Speed": "2791 rpm",
"Mass Air Flow": "17 gr/sec", "Throttle Position": "41 %"}
12 May 13:33:08 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:09 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "12.683985655217327 V", "Temperature": "87.79083526854242 oC", "Engine Speed": "2426 rpm",
"Mass Air Flow": "27 gr/sec", "Throttle Position": "41 %"}
12 May 13:33:09 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:10 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "13.174322516463171 V", "Temperature": "88.43999786836191 oC", "Engine Speed": "2860 rpm",
"Mass Air Flow": "14 gr/sec", "Throttle Position": "24 %"}
12 May 13:33:10 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:11 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "12.266535913324873 V", "Temperature": "89.66635596329192 oC", "Engine Speed": "2747 rpm",
"Mass Air Flow": "11 gr/sec", "Throttle Position": "34 %"}
12 May 13:33:11 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:12 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "13.248804410170937 V", "Temperature": "89.13049396373448 oC", "Engine Speed": "2529 rpm",
"Mass Air Flow": "28 gr/sec", "Throttle Position": "50 %"}
12 May 13:33:12 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:13 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "13.46576810943152 V", "Temperature": "91.82226765196396 oC", "Engine Speed": "2496 rpm",
"Mass Air Flow": "20 gr/sec", "Throttle Position": "60 %"}
12 May 13:33:13 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
12 May 13:33:14 - [info] [moduleoutput:5d5c68f0.643ed8] Sending Message to Azure IoT Edge: Edge0BD2Data
Payload: {"Voltage": "13.274656939934559 V", "Temperature": "88.6977067391218 oC", "Engine Speed": "2539 rpm",
"Mass Air Flow": "20 gr/sec", "Throttle Position": "60 %"}
12 May 13:33:14 - [info] [moduleoutput:5d5c68f0.643ed8] Message sent.
```

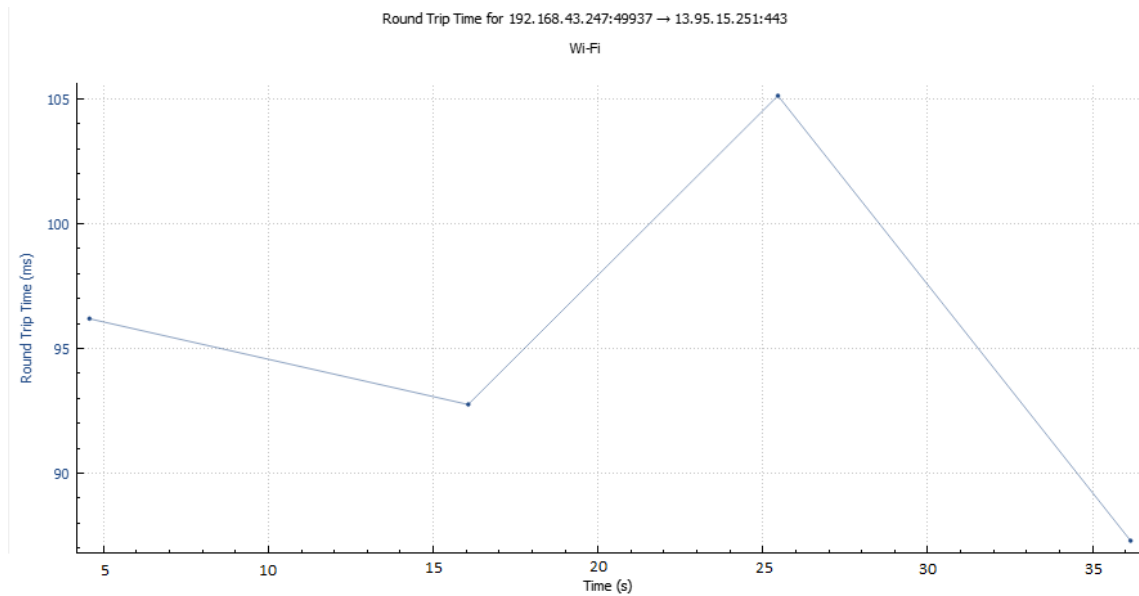
Εικόνα 5.1-2: Μηνύματα τιμών αισθητήρων από συσκευή Edge προς το Hub.

5.2 Πειραματικά αποτελέσματα

Στο υποκεφάλαιο αυτό παρουσιάζονται τα αποτελέσματα των μετρήσεων που εξήχθησαν από τα διαγράμματα του Wireshark.

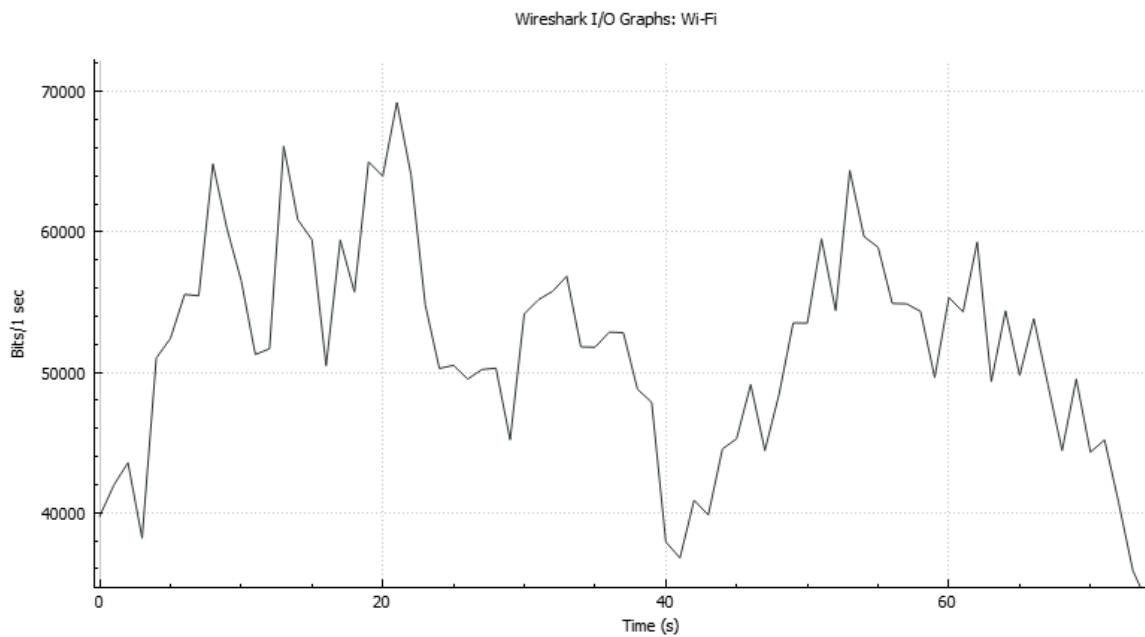
Αποτελέσματα πειράματος 1

Στα Διαγράμματα 5-1 και 5-2 απεικονίζεται η μέτρηση του Round Trip Time και bit/sec αντίστοιχα, κατά την αποστολή μηνυμάτων από τους αισθητήρες στο Core Cloud μέσω της φορητής υπολογιστικής μονάδας, που βρίσκεται σε απόσταση περίπου ενός χιλιομέτρου από το όχημα και δεν «τρέχει» κάποιο σχετικό container σε αυτή.



Διάγραμμα 5-1: RTT από αισθητήρες προς Core Cloud

Στη περίπτωση του Διαγράμματος 5-1 το RTT ξεκινάει από τα 97 ms και μετά τις 3 πρώτες αποστολές παρουσιάζεται αύξηση 8.24% αγγίζοντας τα 105 ms και στη συνέχεια μειώνεται κοντά στα 43 ms.



Διάγραμμα 5-2: I/O graph κατά την μεταφόρτωση από αισθητήρες προς Core Cloud

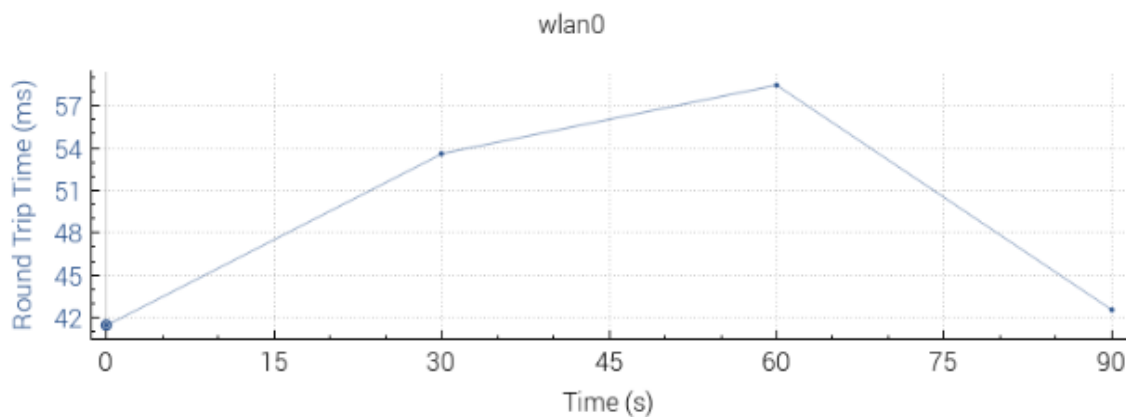
Στο Διάγραμμα 5-2 παρουσιάζεται η κατανάλωση εύρους. Στις 5 πρώτες ανταλλαγές μηνυμάτων παρατηρούνται αυξήσεις και μειώσεις στην κατανάλωση bit/sec περίπου 20%. Μετά την 8^η αποστολή μηνυμάτων η κατανάλωση υφίσταται μια σημαντική

μείωση κάτω από 40000 bit/sec και στη συνέχεια αυξάνεται διατηρώντας τιμές πάνω από 50000 bit/sec.

Αποτελέσματα πειράματος 2

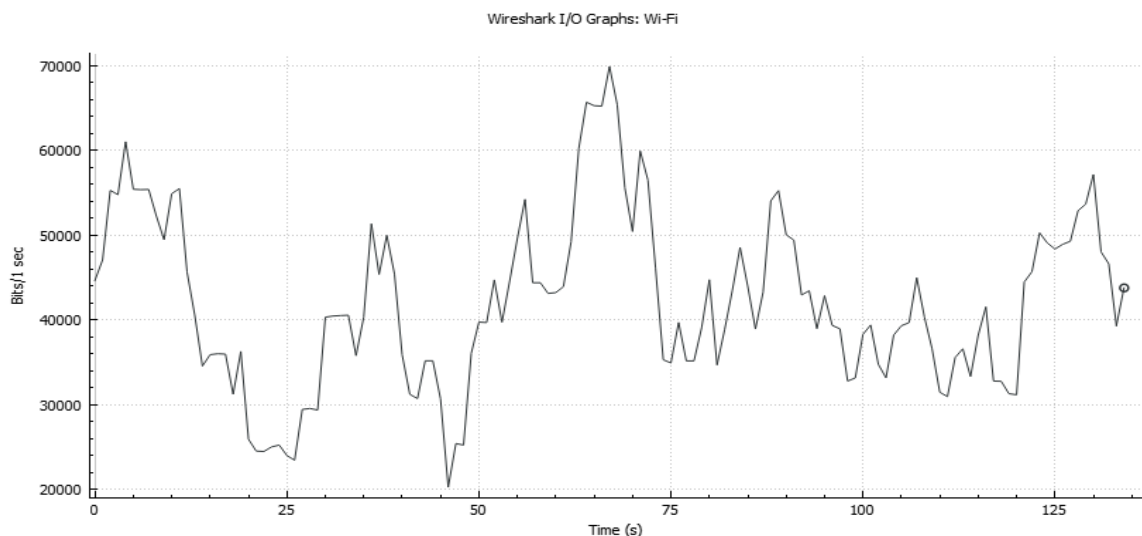
Στα Διαγράμματα 5-3 και 5-4 απεικονίζονται οι τιμές του Round Trip Time και bit/sec αντίστοιχα, κατά την αποστολή μηνυμάτων από τους αισθητήρες στο Core Cloud μέσω της συσκευής Edge (Raspberry Pi) που βρίσκεται εγκατεστημένη στο όχημα και «τρέχει» ο node-RED container σε αυτή.

Round Trip Time for 192.168.43.204:37507 → 13.95.15.251:443



Διάγραμμα 5-3: RTT από συσκευή Edge προς Core Cloud

Στο Διάγραμμα 5-3 το RTT ξεκινάει από 40 ms αυξάνεται περίπου 40% φτάνοντας τα 58 ms και στη συνέχεια μειώνεται και σταθεροποιείται στα 42 ms.



Διάγραμμα 5-4: I/O graph κατά την μεταφόρτωση από συσκευή Edge προς Core Cloud

Στο Διάγραμμα 5-4 στη συσκευή Edge κατά την έναρξη αποστολής μετρήθηκαν τιμές κατανάλωσης bandwidth κοντά στα 45000 bits/sec. Το 67% της κατανάλωσης εύρους

ζώνης κυμαίνεται μεταξύ 30000-50000 bits/sec, ενώ το 33% ξεπερνάει τα 50000 bits/sec.

Σύγκριση αποτελεσμάτων των πειραμάτων 1 και 2

Παρατηρώντας τα Διαγράμματα 5.1 και 5.3 που αφορούν το RTT των 2 πειραμάτων διακρίνεται ότι ο χρόνος καθυστέρησης κατά την έναρξη αποστολής στο Edge είναι σχεδόν 50% μικρότερος από την αποστολή απευθείας στο Core Cloud. Επίσης στο Edge η μέγιστη τιμή που σημειώθηκε ήταν 58 ms, ενώ στο 1^ο πείραμα 105 ms.

Στα Διαγράμματα 5.2 και 5.4 όπου εμφανίζεται η καμπύλη κατανάλωσης εύρους ζώνης, ενώ κατά την έναρξη εμφανίζονται υψηλότερες τιμές στο Edge, στη συνέχεια μειώνονται σημαντικά, μένοντας αρκετές χρονικές στιγμές κάτω από 45000 bits/sec. Στο 1^ο πείραμα, η μέση τιμή χρήσης του bandwidth είναι 54000 bits/sec και στο 2^ο πείραμα η μέση τιμή είναι περίπου 42000 bits/sec, δηλαδή 20% μικρότερη. Έτσι, με την χρήση της Edge συσκευής υπήρξε μια σχετική μείωση, που οφείλεται στην επεξεργασία των καταγεγραμμένων τιμών πριν αποσταλούν στο Core Cloud. Επίσης, στο 2^ο πείραμα ενεργοποιήθηκε εσκεμμένα η άμεση ειδοποίηση με e-mail από 2 αισθητήρες γι' αυτό και παρατηρείται μεγάλη κατανάλωση κατά την έναρξη. Παράλληλα ο χρόνος αποστολής από τη συσκευή Edge στο Core Cloud είχε ρυθμιστεί με περίοδο 1 sec, ενώ κατά την αποστολή από τους αισθητήρες στο Core Cloud στα 5 sec. Η διαφορά στον μέσο όρο φτάνει τα 12000 bits/sec.

Σύμφωνα με όσα αναφέρθηκαν στα Κεφάλαια 3, 4 και 5, στον παρακάτω πίνακα (Πίνακας 5-1) γίνεται μια συγκριτική ανάλυση Cloud και Edge Computing.

Πίνακας 5-1: Συγκριτική ανάλυση Cloud και Edge Computing

Παράμετροι / Περιορισμοί	Νέφος	Άκρο
Μετανάστευση	Ναι	Ναι
Ενοποίηση	Ναι	Ναι
Τοποθέτηση	Ναι	Ναι
Τοποθέτηση περιεχομένου	Όχι	Ναι
Μετανάστευση περιεχομένου	Όχι	Ναι
Περιορισμός πόρων	Όχι	Ναι
Υπερφόρτωση πόρων	Υψηλή	Πολύ υψηλή
Απόσταση από τον τελικό χρήστη	Υψηλή	Χαμηλή

Τύπος δικτύου πρόσβασης	Παραδοσιακός	Υψηλής ταχύτητας
Γεωκατανομή	Ευρείας περιοχής	Τοπικής περιοχής
Βιωσιμότητα	Ναι	Όχι
Κατανάλωση ενέργειας	Υψηλή	Χαμηλή
Συμβατό με το IoT	Ναι	Ναι
Κόστος	Υψηλό	Χαμηλό
Καθυστέρηση μετάδοσης	Μέτρια	Χαμηλή
Εύρος ζώνης	Υψηλό	Χαμηλό

6 Επίλογος

6.1 Σύνοψη και συμπεράσματα

Σε αυτή την εργασία μελετήθηκε ο εξελισσόμενος τομέας του Διαδικτύου των Πραγμάτων και της Υπολογιστικής Νέφους, που σε συνδυασμό με τους τομείς των μεταφορών και των αυτοκινητοβιομηχανιών μπορούν να προσφέρουν πολλά πλεονεκτήματα στις κοινωνίες. Μετά την ανάλυση των λειτουργιών και των αρχιτεκτονικών του IoT και της Υπολογιστικής Νέφους, προτάθηκε και εφαρμόστηκε η υλοποίηση μιας πειραματικής ανάλυσης που αξιοποιεί τις τεχνολογίες αυτές για την παρακολούθηση της καθυστέρησης μετάδοσης και κατανάλωσης εύρους ζώνης, κατά την διάρκεια αποστολής μηνυμάτων από κινούμενο όχημα.

Ακολούθησε η παρουσίαση των αποτελεσμάτων τα οποία ήταν τα αναμενόμενα, όμως παρουσιάζεται και πόσο μεγάλη είναι η βελτίωση προσθέτοντας συσκευή Edge. Η καθυστέρηση μετάδοσης κατά την αποστολή από το Edge στο Core Cloud υπήρξε χαμηλότερη από την προσέγγιση του 1^{ου} πειράματος, μη ξεπερνώντας τα 60 msec, καθότι κατά την αποστολή μηνυμάτων από τους αισθητήρες απευθείας στο Cloud, ως επί το πλείστον, κυμαινόταν μεταξύ 80-95 ms. Έτσι, παρουσιάζεται μια βελτίωση περίπου 35% με την χρήση Edge, ποσοστό σημαντικό στην περίπτωση όπου όλα τα οχήματα με θύρα OBDII είναι συνδεδεμένα με κάποιο Cloud. Η μειωμένη αυτή καθυστέρηση μπορεί να φανεί ιδιαίτερος χρήσιμη στην περίπτωση όπου ο οδηγός χρειάζεται να ειδοποιηθεί για έντονη επιβράδυνση ή σε περίπτωση που το όχημα είναι αυτόνομο να επιβραδύνει άμεσα ή να αλλάξει λωρίδα για αποφυγή εμποδίου. Επίσης η μειωμένη καθυστέρηση μπορεί να φανεί χρήσιμη στην άμεση απενεργοποίηση του κινητήρα ή κάποιου

υποσυστήματος ώστε να αποφευχθεί μεγαλύτερη βλάβη, όταν ο οδηγός δεν έχει αντιληφθεί εγκαίρως την υποκείμενη φθορά. Όσον αφορά την κατανάλωση εύρους ζώνης, κατά την αποστολή από τη συσκευή Edge στο Core Cloud, αν και δημιουργήθηκαν συνθήκες μεγαλύτερης επιβάρυνσης, ο μέσος όρος είναι μικρότερος και η βελτίωση που παρατηρήθηκε κοντά στο 20%. Αυτή η βελτίωση μπορεί αξιοποιηθεί όταν, σε μακροσκοπικό επίπεδο, όλα τα οχήματα θα είναι συνδεδεμένα στο IoT διατηρώντας χαμηλές καταναλώσεις εύρους ζώνης, βελτιώνοντας έτσι την ροή των δεδομένων από τους χρήστες προς το υπολογιστικό νέφος και αντιστρόφως, καθώς και στην μείωση ή αποτροπή απώλειας πακέτων κατά την μετάδοση.

Συμπερασματικά, η υπολογιστική άκρου προσφέρεται για εφαρμογές όπου απαιτούνται πόροι επεξεργασίας δεδομένων και η πηγή των δεδομένων βρίσκεται μακριά από το νέφος αποθήκευσης. Στην αυτοκινητοβιομηχανία, είναι μια καλή και οικονομική λύση, αν εξαιρεθεί το κόστος των υπηρεσιών αποθήκευσης στο Cloud, για παροχή επιπρόσθετης ασφάλειας, καθώς μπορεί να συμβάλει σε συνεργασία με το IoT και την υπολογιστική νέφος στην ενεργητική ασφάλεια του αυτοκινήτου. Επίσης παρέχει μεγαλύτερη κλιμάκωση σε εφαρμογές IoT αλλά και ασφάλεια από κακόβουλους χρήστες του διαδικτύου.

6.2 Όρια και περιορισμοί της έρευνας

Ένας μεγάλος περιορισμός στην ανάπτυξη της εργασίας ήταν το κόστος αγοράς διαφόρων αισθητήρων και κάμερας για την καταγραφή και αποστολή στο Cloud μεγαλύτερου όγκου δεδομένων. Έτσι έγινε χρήση ενός OBD reader χαμηλού κόστους με δυνατότητα ανάγνωσης τιμών από έξι αισθητήρες που προϋπάρχουν στο όχημα. Επίσης όσον αφορά την άμεση απόφαση σε περίπτωση πιθανής βλάβης, η δυνατότητα που υπήρχε από το node-RED ήταν να σταλεί κάποια ειδοποίηση με email όταν οι τιμές των αισθητήρων παρεκκλίνουν από τις αναμενόμενες. Ακόμη, όσον αφορά την απόδοση της επικοινωνίας του οχήματος με το IoT Hub όταν διακόπτεται το σήμα από το 4G access point του smartphone, λόγω της αρχιτεκτονικής που χρησιμοποιήθηκε, σταματάει η αποστολή δεδομένων στο cloud και η αποστολή επανεκκινεί όταν έλθει το όχημα εντός δικτύου έχοντας χάσει όμως τα δεδομένα όσο βρισκόταν εκτός δικτύου. Η ανάλυση των καταγεγραμμένων δεδομένων μέσα από μια βάση δεδομένων έμεινε εκτός της εργασίας διότι κατά βάση έγινε μελέτη καθυστέρησης και χρήσης εύρους ζώνης σε επίπεδο δικτύου.

6.3 Μελλοντικές Επεκτάσεις

Το σενάριο που αναπτύχθηκε στην συγκεκριμένη εργασία είναι εύκολα υλοποιήσιμο και επεκτάσιμο. Μελλοντικά θα μπορούσαν να προστεθούν περισσότεροι αισθητήρες και ενεργοποιητές όπως κάμερες περιμετρικά του οχήματος, επιπλέον αισθητήρες θερμοκρασίας λέιζερ για την μέτρηση της θερμοκρασίας των ελαστικών, επιταχυνσιόμετρα και αισθητήρες δονήσεων σε διάφορα εξαρτήματα όπως παραδείγματος χάριν οι αναρτήσεις, με σκοπό την παρακολούθηση του εύρους ζώνης και της καθυστέρησης μετάδοσης όταν το μέγεθος των δεδομένων αυξάνεται σε επίπεδο Μεγάλων Δεδομένων. Αυτό θα εξυπηρετούσε επίσης τις αυτοκινητοβιομηχανίες και εταιρίες κατασκευής εξαρτημάτων στην βελτίωση των προϊόντων τους στην περίπτωση που θα μπορούν να έχουν πρόσβαση στα δεδομένα αυτά. Επίσης θα μπορούσε να μελετηθεί η ανάπτυξη μοντέλων πρόβλεψης, σε οδικό επίπεδο, στο Edge Cloud με σκοπό την μείωση του λάθους από τον χρήστη και την άμεση αντίδραση του οχήματος με την βοήθεια ενεργοποιητών. Επιπρόσθετα, θα ήταν ενδιαφέρον η ανάπτυξη παρόμοιας εφαρμογής σε γλώσσα python έτσι ώστε να εξαλειφθεί η καθυστέρηση μετάδοσης που υπάρχει για την διαδρομή των δεδομένων από την συνεργασία της εφαρμογής Torque και του node-RED.

Βιβλιογραφία

- Abdel Baset, M., Manogaran, G., Mohamed, M. and Rushty, E., 2018. Internet of things in smart education environment: Supportive framework in the decision making process. *Concurrency and Computation Practice and Experience*, May 2018.
- Atzoria, L., Ierab, A. and Morabito G., 2010. The Internet of Things: A survey, *Computer Networks*, vol. 54 no.2010, pp. 2787–2805, June 2010.
- Awang, A., Husain, K., Kamel, N. and Aïssa, S., 2017. Routing in vehicular adhoc networks: A survey on single and cross-layer design techniques, and perspectives. *IEEE Access*, vol. 5, pp. 9497–9517.
- Ben-Daya, M., Hassini, E. and Bahroun, Z., 2019. Internet of things and supply chain management: a literature review. *International Journal of Production Research*, vol. 57, no. 15–16, pp. 4719–4742.
- Burhan, M., Rehman, R.A., Khan B. and Kim, B.S., 2018. IoT Elements, Layered Architectures and Security Issues: A Comprehensive Survey. *Sensors* 18, 2796.
- Campbell, M., Egerstedt, M., How, J.P. and Murray, R.M., 2010. Autonomous driving in urban environments: approaches, lessons and challenges. *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences*, vol. 368 no. 1928, pp. 4649-4672, October 2010.
- Cheng, G.J., Liu, L.T., Qiang, X.J. and Liu, Y., 2016. Industry 4.0: Development and Application of Intelligent Manufacturing. Conference: 2016 International Conference on Information System and Artificial Intelligence (ISAI), June 2016
- Contreras, J., Zeadally, S. and Guerrero-Ibanez, J.A., 2017. Internet of vehicles: Architecture, protocols and security. *IEEE, Internet of Things Journal*, PP(99):1-1, April 2017.
- Darwish, T. and Bakar, K.A., 2015. Traffic density estimation in vehicular ad hoc networks: A review. *Ad Hoc Networks*, vol. 24, pp. 337–351, January 2015.
- Darwish, T.S.J. and Bakar, K., 2018. Fog Based Intelligent Transportation Big Data Analytics in The Internet of Vehicles Environment: Motivations, Architecture, Challenges and Critical Issues. *IEEE Access*, vol. 6, pp. 15679-15701, April 2018.
- Deng, R., Lu, R., Lai, C., Luan, T.H. and Liang, H., 2016. Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption. *IEEE, Internet of Things Journal*, vol. 3, no. 6, pp. 117-1181, December 2016.

Docker, 2020. What is a Container? [online] Available at: <<https://www.docker.com/resources/what-container>>

[Accessed 24 March 2020].

Dorsemaine, B., Gaulier, J.P., Wary, J., Kheir, N. and Urien, P., 2015. Internet of Things: A Definition & Taxonomy, 9th International Conference on Next Generation Mobile Applications, Services and Technologies.

Duta, S., 2016. Security Issues in Cloud Computing. *International Journal of Emerging Trends & Technology in Computer Science*, vol. 4, no. 11, pp. 4747-4752, November 2016.

Fangchun, Y., Shangguang, W., Jinglin, L., Zhihan, L. and Qibo, S., 2014. An overview of Internet of Vehicles. *China Communications*, vol. 11, no. 10, pp. 1–15, October 2014.

Felter, W., Ferreira, A., Rajamony, R. and Rubio, J., 2015. An Updated Performance Comparison of Virtual Machines and Linux Containers. *IEEE, International Symposium on Performance Analysis of Systems and Software*, vol. 25482, pp. 171–172.

Fielding, R., Irvine, U.C., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P. Berners-Lee, T., 1999. Method Definitions. *The internet society, RFC 2616*, pp. 51–57, sec. 9.

Fielding, R., Roy, T., Gettys, J., Mogul, J.C., Nielsen, H.F., Masinter, L., Leach, P.J. and Berners-Lee, T., 1999. *Hypertext Transfer Protocol-HTTP/1.1. IETF, RFC 2616*.

Fuqaha, A.A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M., 2015. Internet of Things: A Survey on Enabling Technologies, Protocols and Applications. *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, November 2015.

Golestan, K., Soua, R., Karray, F. and Kamel, M.S., 2016. Situation awareness within the context of connected cars: A comprehensive review and recent trends. *Information Fusion*, vol. 29, pp. 68–83, May 2016.

Gross, M., 2016. A planet with two billion cars. *Current Biology*, vol. 26, Issue 8, pp. R307–R310, April 2016.

Gunarathne, T., Zhang, B., Wu, T.L. and Qiu, J., 2013. Scalable parallel computing on clouds using twister4azure iterative mapreduce. *Future Generation Computer Systems*, vol. 29, no. 4, pp. 1035–1048, June 2013.

Islam, A., Hossan, M.T. and Jang, Y.M., 2018. Convolutional neural network scheme–based optical camera communication system for intelligent Internet of vehicles. *International Journal of Distributed Sensor Networks*, Vol. 14, no.4, February 2018.

Kagermann, H., Walhster, W. and Helbig, J., 2013. Recommendations for implementing the strategic initiative industry 4.0, acatech, National Academy of Science and Engineering, April 2013.

Kaiwartya, O., Abdullah, A.H., Cao, Y., Altameem, A., Prasad, M., Lin, C.T. and Liu X., 2016. Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects. *IEEE*, vol. 4, pp. 5356–5373, September 2016.

Kumar, S., Gollakota, S. and Katabi, D., 2012. A Cloud-Assisted Design for Autonomous Driving. Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, August 2012, Helsinki, Finland. ACM (2012).

Lee, E.K., Gerla, M., Pau, G., Lee, U. and Lim, J.H., 2016. Internet of vehicles: From intelligent grid to autonomous cars and vehicular fogs. *International Journal of Distributed Sensor Networks*, vol. 12, no. 9, September 2016.

Li, S., Xu, L.D. and Zhao, S., 2015. The internet of things: A survey. *Information Systems Frontiers*, vol 17, no. 2, pp. 243-259, January 2015

Linthicum, D., 2019. Edge computing vs fog computing: Definitions and enterprise uses. [online] Available at: <<https://www.cisco.com/c/en/us/solutions/enterprise-networks/edge-computing.html>> [Accessed 15 March 2020].

Lu, C.W., Hsieh, C.M., Chang, C.H. and Yang, C.T., 2013. An improvement to data service in cloud computing with content sensitive transaction analysis and adaptation. *IEEE, 37th Annual Computer Software and Applications Conference Workshops*, pp. 463–468, July 2013.

Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Abaker, I., Hashem, T., Siddiq, A. and Yaqoob I., 2017. Big IoT Data analytics: Architecture, opportunities and open research challenges. *IEEE Access*, vol. 5, pp. 5247-5261, March 2017.

Maurelli, F., Droschel, D., Wisspeintner, T., May S. and Surmann, H., 2009. A 3D laser scanner system for autonomous vehicle navigation. *International Conference on Advanced Robotics*.

Mell, P. and Grance, T., 2011. The NIST definition of cloud computing. National Institute of Standards and Technology, Tech. Rep. 800-145.

Miorandi, D., Sicari, S., Pellegrini, F.D. and Chlamtac, I., 2012. Internet of Things: Vision, Applications and Research Challenges. *Ad Hoc Networks* 10, September 2012.

Mukherjee, M., Matam, R., Shu, L., Maglaras, L., Ferrag, M.A., Choudhury, N. and Kumar, V., 2017. Security and privacy in fog computing: Challenges. *IEEE Access*, vol. 5, pp. 19293-19304.

Nabeena, A. and Rahman, B.S.A., 2015. A review on cloud based patient monitoring. *International Journal of Information Technology & Systems*, vol. 4, no. 1, ISSN: 2277-9825, January-June 2015.

Naranjo, P.G.V., Shojafar, M., Vaca-Cardenas, L., Canali, C., Lancellotti, R. and Baccarelli, E., 2016. Big data over smartgrid - a fog computing perspective. *SOFTCOM Workshop*, pp. 1-6.

Nitti, M., Girau, R., Floris, A. and Atzori, L., 2014. On adding the social dimension to the Internet of Vehicles: Friendship and middleware. *IEEE, International Black Sea Conference on Communications and Networking*, pp. 134–138, May 2014.

Priyadarsiny, U., 2018. Real World IoT Applications in Different Domains. [online] Available at: <<https://edureka.co/iot-applications>> [Accessed 7 March 2020].

Rose, K., Eldridge, S. and Chapin, L., 2015. The Internet of Things: An overview, *Internet Society*, October 2015.

SUPINFO International University, 2020. Internet of Things. [online] Available at: <<https://www.supinfo.com/articles/single/4235-internet-of-things>> [Accessed 12 March 2020].

Tang, B., Chen, Z., Hefferman, G., Pei, S., Wei, T., He, H. and Yang, Q., 2017. Incorporating intelligence in fog computing for big data analysis in smart cities. *IEEE, Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2140-2150, October 2017.

Tsai, C.W., Lai, C.F., Chao, H.C and Vasilakos, A.V., 2015. Big data analytics: A survey. *Journal of Big Data*, vol. 2, no. 1, p. 21.

Vallati, C., Viridis, A., Mingozi, E. and Stea, G., 2016. Mobile-edge computing come home connecting things in future smart homes using LTE device to device communications. *IEEE, Consumer Electronics Magazine*, vol. 5, no. 4, pp. 77-83, April 2016.

Wikipedia contributors, 2020. Bluetooth. [online] Wikipedia. Available at: <<https://el.wikipedia.org/wiki/Bluetooth>> [Accessed 10 March 2020].

Wikipedia contributors, 2020. Internet of Things. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/Internet_of_things>

[Accessed 8 March 2020].

Wikipedia contributors, 2020. On-Board Diagnostics. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/On-board_diagnostics#OBD-I>

[Accessed 10 March 2020].

Wikipedia contributors, 2020. Raspberry Pi OS. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/Raspberry_Pi_OS>

[Accessed 12 March 2020].

Wikipedia contributors, 2020. Raspberry Pi. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/Raspberry_Pi>

[Accessed 8 March 2020].

Wikipedia contributors, 2020. Tethering. [online] Wikipedia. Available at: <<https://en.wikipedia.org/wiki/Tethering>>

[Accessed 30 March 2020].

Wikipedia contributors, 2020. Wi-Fi. [online] Wikipedia. Available at: <<https://en.wikipedia.org/wiki/Wi-Fi>>

[Accessed 23 March 2020].

Wikipedia contributors, 2020. Wireless LAN. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/Wireless_LAN>

[Accessed 12 March 2020].

Wortmann, F. and Flüchter, K., 2015. Internet of Things. *Bus. Inf. Syst. Eng.*, vol. 57, no. 3, pp. 221-224.

Xiao, Y. and Zhu, C., 2017. Vehicular fog computing: Vision and challenges. *IEEE, International Conference on Pervasive Computing and Communications Workshops*, vol. 10, pp. 6-9, May 2017.

Xu, L.D., Wu, H. and Shancang, L., 2014. Internet of Things in Industries: A Survey. *IEEE, Transactions on industrial informatics*, vol.10, no.4, November 2014.

Xu, W., Zhou, H., Cheng, N., Lyu, F., Shi, W., Chen, J. and Shen, X., 2018. Internet of Vehicles in Big Data Era. *IEEE/CAA, Journal of Automatica Sinica*, vol. 5, no. 1, January 2018.

Yu, W., Liang, F., He, X., Hatcher, W.D., Lu, C., Lin, J. and Yang, X., 2017. A Survey on the Edge Computing for the Internet of Things, IEEE, vol. 6, pp. 6900-6919, November 2017.

Zhou, K., Liu, T. and Zhou, L., 2015. Industry 4.0: Towards future industrial opportunities and challenges. Computer Science, 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Published 2015.