

Τίτλος: Εφαρμογές ICN σε διαδικτυακή και διαστημική τεχνολογία
Έκθεση Αναφοράς της Καπετανίδου Ιωάννας-Αγγελικής

Κωδικός Έργου: 82120

Επιστημονικός Υπεύθυνος: Τσαουσίδης Βασίλειος

Πίνακας περιεχομένων

Σκοπός Μελέτης	4
Κεφάλαιο 1 ^ο : Εισαγωγή.....	5
1.1 Πληροφοριοκεντρική Δικτύωση (Information-Centric Networking (ICN))	5
1.2 Μια επισκόπηση της αρχιτεκτονικής Named-Data Networking (NDN).....	6
1.2.2 Ασφάλεια και εμπιστοσύνη στο NDN	8
1.3 Edge Computing	9
Κεφάλαιο 2 ^ο : Μεθοδολογία.....	11
2.1 Γενικές Παράμετροι και Μετρικές.....	11
2.2 Σενάρια.....	12
2.3 Τοπολογία Δικτύου.....	12
Κεφάλαιο 3 ^ο : Μηχανισμός	16
3.1 1 ^η Έκδοχή.....	16
3.2 2 ^η Έκδοχή.....	18
Κεφάλαιο 4 ^ο : Αποτελέσματα.....	21
4.1 Σύγκριση της Απόδοσης (Throughput).....	21
4.1.1 Απόδοση τα πρώτα 100 secs της προσομοίωσης	21
4.1.2 Απόδοση μετά τα πρώτα 100 secs της προσομοίωσης	23
4.2 Σύγκριση μέσης Απόδοσης.....	25
Κεφάλαιο 5 ^ο : Συμπεράσματα.....	28
5.1 Γενικά συμπεράσματα	28
5.2 Μελλοντικές Κατευθύνσεις.....	28
Βιβλιογραφία	30

Πίνακας εικόνων

Εικόνα 1: Η αρχιτεκτονική NDN	6
Εικόνα 2: Βασικές Δομές ενός NDN router	7
Εικόνα 3: Επεξεργασία Interest Packet από έναν NDN router	8
Εικόνα 4: Επεξεργασία Data Packet από έναν NDN router	8
Εικόνα 5: Τοπολογία 10 ζευγών	13
Εικόνα 6: Τοπολογία 20 ζευγών	14
Εικόνα 7: Τοπολογία 25 ζευγών	14
Εικόνα 8: Τοπολογία 50 ζευγών	15
Εικόνα 9: Σύγκριση των Throughput στη ρίζα για 10 καταναλωτές τα 100 πρώτα δευτερόλεπτα.....	21
Εικόνα 10: Σύγκριση των Throughput στη ρίζα για 20 καταναλωτές τα 100 πρώτα δευτερόλεπτα.....	21
Εικόνα 11: Σύγκριση των Throughput στη ρίζα για 25 καταναλωτές τα 100 πρώτα δευτερόλεπτα.....	22
Εικόνα 12: Σύγκριση των Throughput στη ρίζα για 50 καταναλωτές τα 100 πρώτα δευτερόλεπτα.....	22
Εικόνα 13: Σύγκριση Throughput στη ρίζα για 10 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα.....	23
Εικόνα 14: Σύγκριση Throughput στη ρίζα για 20 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα.....	23
Εικόνα 15: Σύγκριση Throughput στη ρίζα για 25 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα.....	24
Εικόνα 16: Σύγκριση Throughput στη ρίζα για 50 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα.....	24
Εικόνα 17: Σύγκριση μέσου Throughput στη ρίζα για 10 καταναλωτές.....	25
Εικόνα 18: Σύγκριση μέσου Throughput στη ρίζα για 20 καταναλωτές.....	26
Εικόνα 19: Σύγκριση μέσου Throughput στη ρίζα για 25 καταναλωτές.....	26
Εικόνα 20: Σύγκριση μέσου Throughput στη ρίζα για 50 καταναλωτές.....	27

Σκοπός Μελέτης

Μέχρι σήμερα, στα δίκτυα υπολογιστών βρίσκεται εφαρμογή η Host-Centric αρχιτεκτονική, βασικός κανόνας της οποίας είναι η χρήση διευθύνσεων ως αναγνωριστικά. Με άλλα λόγια, κάθε host (συσκευή, χρήστης ή server) που είναι μέλος ενός δικτύου ταυτοποιείται από μια μοναδική IP διεύθυνση. Η πληροφορία σ' αυτή την αρχιτεκτονική είναι άρρηκτα συνδεδεμένη με την φυσική τοποθεσία αποθήκευσής της (π.χ. έναν κεντρικό server). Ωστόσο, η ισχύουσα αρχιτεκτονική αδυνατεί να ανταπεξέλθει στο συνεχώς αυξανόμενο αριθμό χρηστών και εφαρμογών που απαιτούν την ταχύτερη ανάκτηση των ζητούμενων δεδομένων και στην επιβάρυνση του δικτύου με πληθώρα πληροφοριών. Η προκύπτουσα ανάγκη για μια ριζική αλλαγή στο σχεδιασμό των δικτύων ήταν το κίνητρο για το σχεδιασμό της πληροφοριοκεντρικής δικτύωσης (Information-Centric Networking (ICN)) με κυρίαρχο παράδειγμα την αρχιτεκτονική Named Data Networking (NDN) [1]. Η λογική αυτής της αρχιτεκτονικής περιστρέφεται γύρω από το την εστίαση στο περιεχόμενο της ίδιας της πληροφορίας και την απόδοση ονομάτων στα πακέτα δεδομένων. Έτσι η πληροφορία αποδεσμεύεται από μια κεντρική θέση αποθήκευσης και οι χρήστες πλέον αναζητούν ακριβώς αυτό που τους ενδιαφέρει αντί για τη διεύθυνσή του.

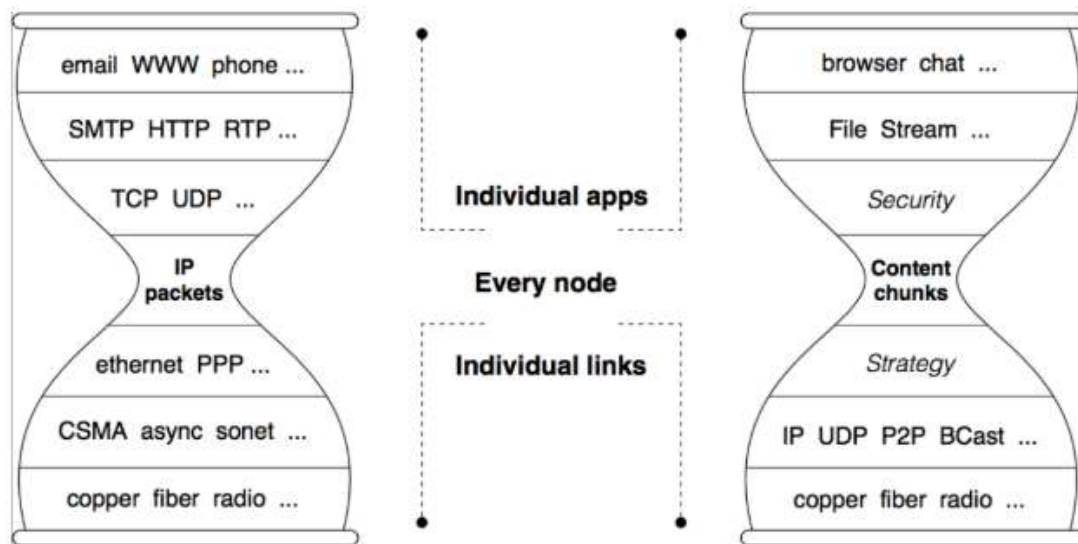
Η μελέτη που παρουσιάζεται εδώ έχει διεκπεραιωθεί στα πλαίσια της μελλοντικής αρχιτεκτονικής NDN, με τη βοήθεια του προσομοιωτή ndnSIM. Πιο συγκεκριμένα, εστιάζει στον τομέα της αξιοπιστίας των δικτύων προτείνοντας ένα μηχανισμό για την αξιολόγηση των διακινούμενων πακέτων σε ένα δίκτυο και συνεπώς των χρηστών που τα παρέχουν. Η εφαρμογή υλοποιείται σε δύο εκδοχές που διαφέρουν ως προς την επεξεργασία των δεδομένων που συλλέγονται στους κόμβους του δικτύου: η πρώτη ακολουθεί τη μέθοδο του Cloud Computing, ενώ η δεύτερη βασίζεται στην καινοτόμα ιδέα του Edge Computing. Απώτερος σκοπός είναι η προσαρμογή και χρήση του μηχανισμού σε διαστημικά περιβάλλοντα για την αξιολόγηση κόμβων που μεταφέρουν διαστημικά δεδομένα σε επίγειες βάσεις.

1.1 Πληροφοριοκεντρική Δικτύωση (Information-Centric Networking (ICN))

Στην αρχική του μορφή, το διαδίκτυο είχε σχεδιαστεί για αισθητά λιγότερο αριθμό χρηστών, εφαρμογών και συσκευών απ' ό,τι σήμερα. Για να είναι αποδοτικό αρκούσε να εξασφαλίζει απομακρυσμένη πρόσβαση σε ένα κεντρικό υπολογιστή και να μεταφέρει πακέτα δεδομένων από ένα χρήστη ή server σε ένα άλλο (end-to-end). Αυτή η υλοποίηση αναγκάζει την αρχιτεκτονική του διαδικτύου να εστιάσει σε κεντρικά σημεία/hosts (Host-Centric Networking) και προϋποθέτει πως κάθε συσκευή με πρόσβαση στο διαδίκτυο διαθέτει μια μοναδικά κατοχυρωμένη διεύθυνση IP ως αναγνωριστικό. Αργότερα, με τη συγχρονισμένη εμφάνιση του Παγκόσμιου Ιστού (World Wide Web) και των πολυμέσων, το διαδίκτυο αναγκάζεται να στραφεί στην αποθήκευση πληροφοριών. Όταν ένας χρήστης αναζητά μια πληροφορία απευθύνεται στη διεύθυνση/συσκευή όπου αυτή είναι αποθηκευμένη. Ο σημερινός αυξημένος αριθμός χρηστών απαιτεί να αποκτά τα δεδομένα που αιτείται όσο το δυνατόν ταχύτερα. Αυτό σε συνδυασμό με τον τεράστιο όγκο διαθέσιμων πληροφοριών συνεπάγονται πως χρειάζεται να μεταβούμε από τη λογική που εστιάζει στη φυσική τοποθεσία αποθήκευσης, σε μια λογική που εστιάζει στο περιεχόμενο της πληροφορίας. Εισάγεται, έτσι, η έννοια της αποκεντροποίησης αφού αντί η επικοινωνία και η προώθηση πακέτων να γίνεται αποκλειστικά από μια κεντρική συσκευή σε μια τερματική, πλέον διαμορφώνεται από έναν-σε-πολλούς ή από πολλούς-σε-έναν ή από πολλούς-σε-πολλούς κόμβους. Οι προσπάθειες υλοποίησης αυτής της νέας λογικής στηριζόμενοι στην παραδοσιακή Host-Centric αρχιτεκτονική έχουν επιφέρει μεγάλη πολυπλοκότητα στο δίκτυο και έχουν αποδειχθεί μη αποδοτικές [2].

Όλα αυτά έχουν οδηγήσει την ερευνητική κοινότητα στη στροφή προς την πληροφοριοκεντρική δικτύωση (Information-Centric Networking (ICN)). Σε αντίθεση με το Host-Centric Networking, το ICN προτείνει μια διαφορετική προσέγγιση, προσανατολισμένη στα δεδομένα. Μια κυρίαρχη προτεινόμενη πληροφοριοκεντρική αρχιτεκτονική είναι το Named Data Networking (NDN) [1]. Θεμέλιο λίθο για το NDN αποτελούν τα Named Data Objects (NDO). Σε αυτή την περίπτωση, ένας ενδιαφερόμενος χρήστης κατονομάζει απευθείας τα δεδομένα που αναζητά, χωρίς να τον ενδιαφέρει το πού βρίσκονται αποθηκευμένα. Τα NDO είναι ανεξάρτητα από την

τοποθεσία, την αποθηκευτική μέθοδο και την εφαρμογή που τα χρησιμοποιεί. Αυτό σημαίνει ότι ένα NDO διατηρεί το όνομά του, και ως εκ τούτου την ταυτότητά του, ανεξάρτητα από τη θέση του και από τον τρόπο αντιγραφής, αποθήκευσης και μεταφοράς. Σημαίνει επίσης ότι οποιαδήποτε δύο αντίγραφα ενός NDO είναι ισοδύναμα. Για παράδειγμα, οποιοσδήποτε κόμβος που κατέχει ένα αντίγραφο μπορεί να το παραδώσει σε ένα ενδιαφερόμενο χρήστη [3]. Επιπλέον, με την αρχιτεκτονική Named Data Networking επιτυγχάνεται η προσωρινή αποθήκευση δεδομένων εντός του δικτύου (in-network caching), που διευκολύνει και επιταχύνει την ανάκτηση δεδομένων και την ταυτόχρονη επικοινωνία πολλαπλών κόμβων μέσω μοντέλων αναμετάδοσης και αλληλεπίδρασης [3], [4], [5].



Εικόνα 1: Η αρχιτεκτονική NDN

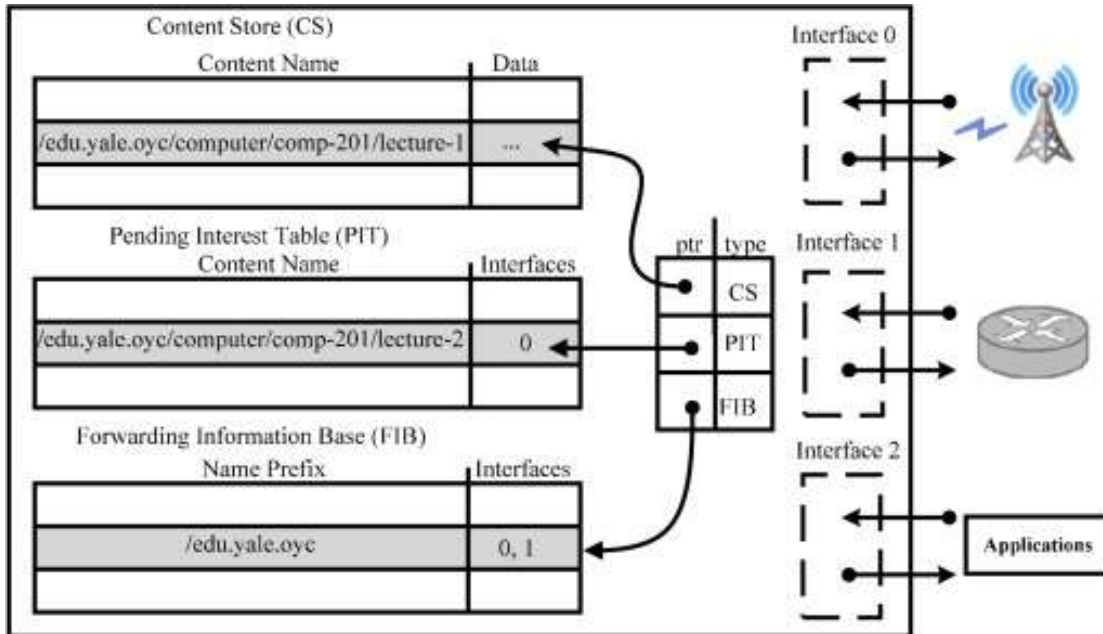
1.2 Μια επισκόπηση της αρχιτεκτονικής Named-Data Networking (NDN)

Στην NDN αρχιτεκτονική, Ένας καταναλωτής (consumer) που επιθυμεί να αποκτήσει ένα πακέτο δεδομένων (Data Packet) εκδηλώνει το ενδιαφέρον στέλνοντας στο δίκτυο ένα Interest Packet στο οποίο συμπεριλαμβάνεται το όνομα του περιεχομένου που τον ενδιαφέρει.

Επιπλέον, κάθε router του δικτύου διατηρεί τρεις βασικές δομές:

- Το Content Store (CS) που λειτουργεί ως προσωρινή μνήμη (cache) όπου αποθηκεύονται τα πιο προσφάτως ληφθέντα πακέτα δεδομένων.

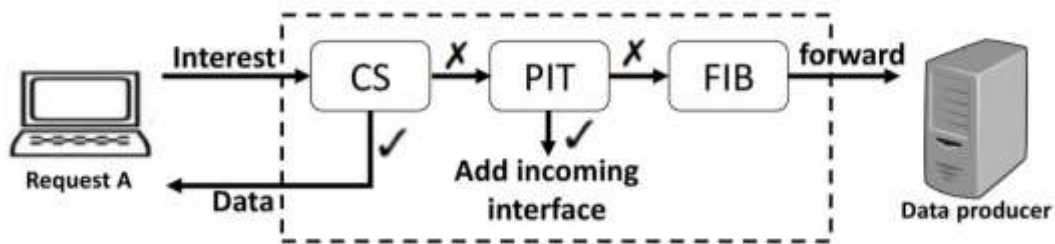
- Το Pending Interest Table (PIT) όπου το router αποθηκεύει όλα τα Interests των οποίων η απάντηση εκκρεμεί, καθώς και τις διεπαφές (interfaces) από τις οποίες τα έλαβε, και
- Τη Forwarding Interest Base (FIB) που περιέχει πληροφορίες για τη δρομολόγηση των Interests.



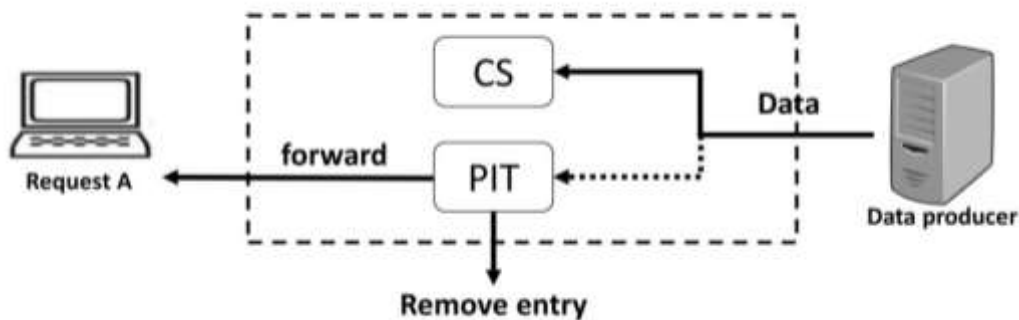
Εικόνα 2: Βασικές Δομές ενός NDN router

Κάθε router από το οποίο περνάει το Interest πρώτα ελέγχει εάν το αντίστοιχο Data Packet είναι διαθέσιμο στο Content Store του. Εάν δεν υπάρχει εκεί αποθηκευμένο το επιθυμητό Data Packet, τότε ελέγχει το Pending Interest Table (PIT). Εάν φτάσουν στο router περισσότερα του ενός Interests για το ίδιο Data Packet, τότε αυτός προωθεί μονάχα το πρώτο, αποθηκεύει όμως στο PIT όλες τις εμπλεκόμενες διεπαφές. Η προώθηση των Interest Packets σε επόμενους κόμβους γίνεται με βάση πληροφορίες που αποθηκεύουν οι routers στην τελευταία δομή, το Forwarding Interest Base (FIB). Μόλις το Interest Packet φτάσει σε κάποιο κόμβο ο οποίος διαθέτει αντίγραφο του ζητούμενου Data Packet, τότε ο κόμβος απαντάει στέλνοντας το. Το Data Packet ακολουθεί την αντίστροφη διαδρομή για να φτάσει στον καταναλωτή. Ας σημειωθεί πως στο χειρίστο σενάριο το Interest θα ταξιδέψει σε όλη τη διαδρομή μέχρι τον παραγωγό (producer) του Data Packet. Προκύπτει, δηλαδή, πως η NDN αρχιτεκτονική δεν απορρίπτει την end-to-end επικοινωνία, απλώς διευρύνει τις δυνατότητές της αξιοποιώντας και τους ενδιάμεσους κόμβους. Στην περίπτωση των πολλαπλών

Interests, όταν το Data Packet φτάσει στο router με το PIT τότε αυτός το στέλνει σε όλους τους καταναλωτές που έχει στη λίστα του. Κάθε καταναλωτής που ικανοποιείται διαγράφεται από τη λίστα. Μόλις απαντηθούν όλα τα Interests ο router αφαιρεί και το Interest και αποθηκεύει το Data Packet που μόλις έλαβε στο Content Store (CS) του. Τα παραπάνω αποτυπώνονται διαγραμματικά ως εξής:



Εικόνα 3: Επεξεργασία Interest Packet από έναν NDN router



Εικόνα 4: Επεξεργασία Data Packet από έναν NDN router

1.2.2 Ασφάλεια και εμπιστοσύνη στο NDN

Η NDN αρχιτεκτονική έχει ασφαλίσει τη λειτουργία της με τη σχεδίαση Trust Schemas. Οι κανόνες που τα διέπουν βασίζονται κυρίως στην υπογραφή των Data Packets. Τα πακέτα υπογράφονται από τους παραγωγούς με κωδικοποίηση δημοσίου κλειδιού. Οι καταναλωτές, από την άλλη, θα πρέπει να έχουν στη διάθεσή τους τα αντίστοιχα

δημόσια κλειδιά για να αποκρυπτογραφήσουν την υπογραφή και να επιβεβαιώσουν την προέλευση του πακέτου. Το ποιο δημόσιο κλειδί χρειάζεται για κάθε πακέτο γίνεται φανερό στους καταναλωτές μέσω αναφορών σε κλειδιά που δημοσιοποιούν οι παραγωγοί. Αυτές τις αναφορές οι καταναλωτές μπορούν, επίσης, να τα διασταυρώσουν με το Data Packet για να εξακριβώσουν πως την εγκυρότητα της υπογραφής για το συγκεκριμένο πακέτο [6].

Αν και με αυτές τις προδιαγραφές οι καταναλωτές νιώθουν σιγουριά επειδή μπορούν να διαπιστώσουν πως ο παραγωγός που υπέγραψε το πακέτο που λαμβάνουν είχε την εξουσιοδότηση να το κάνει, δεν έχουν καμιά πληροφόρηση σχετικά με το πόσο έμπιστος είναι αυτός και για την πιστότητα των πακέτων του. Μια ακόμη βασική αδυναμία του NDN είναι πως αν και τελικά οι χρήστες μπορούν να αναγνωρίσουν ένα μη έγκυρο ή αλλοιωμένο πακέτο, οι routers δεν έχουν κανένα μέσο για να διαπιστώσουν πως ένα πακέτο δεδομένων είναι έγκυρο ή ακέραιο. Αυτό τους καθιστά επιρρεπείς σε νέες μορφές επιθέσεων [7].

1.3 Edge Computing

Το Edge Computing είναι μια πολλά υποσχόμενη τεχνική που έρχεται να διαδεχθεί το Cloud Computing. Στο Cloud Computing αναφερόμαστε τόσο στις εφαρμογές που λειτουργούν ως υπηρεσίες μέσω διαδικτύου όσο και στο hardware και το λογισμικό στα κέντρα που είναι υπεύθυνα γι' αυτές τις υπηρεσίες. Τα κέντρα που ασχολούνται με το hardware και το software είναι αυτά που παρομοιάζονται με cloud, ενώ οι υπηρεσίες που παρέχονται αποτελούν το computing. Οι υπηρεσίες αυτές καλούνται επίσης Software as a Service (SaaS) και είναι ιδιαίτερα διαδεδομένες στους χρήστες του διαδικτύου [8]. Γνωστά παραδείγματα είναι το Facebook, το Twitter κλπ. Περνώντας όμως στην εποχή του Internet of Things (IoT) και των αισθητήρων, εμφανίζεται η ανάγκη να μετέλθουμε στο Edge Computing ώστε να διατηρηθεί η αποδοτικότητα του δικτύου.

Πλέον οι κόμβοι που κάποτε απλά κατανάλωναν πληροφορίες τώρα τείνουν να τις παράγουν. Η συλλογή δεδομένων γίνεται από κάθε χρήστη που διαθέτει μια συσκευή όπως smartphone, smartwatch κλπ. Τα δεδομένα αυτά ποικίλουν από την καταγραφή ενός στιγμιότυπου ή ενός βίντεο, μέχρι τις μετρήσεις διανυμμένης απόστασης ή

βημάτων, των καρδιακών παλμών και άλλα. Όλα αυτά όμως απαιτούν την παραχώρηση μεγαλύτερης υπολογιστικής δύναμης στις έξυπνες συσκευές. Αυτές έχουν τη δυνατότητα να επεξεργαστούν μόνες τους τις πληροφορίες που παράγουν άμεσα, χωρίς να χρειάζεται τη μεσολάβηση του cloud.

Είναι προφανές πως θα ήταν πολύ πιο δαπανηρό για το δίκτυο να στέλνονται όλα τα παραγόμενα δεδομένα από εκατομμύρια συσκευές ταυτόχρονα στο cloud για να τα επεξεργαστεί και να επιστρέψει τα αποτελέσματα στην κάθε συσκευή. Μια τέτοια διαδικασία θα κόστιζε πολύ στην υπερφόρτιση του διαθέσιμου εύρους καθώς και στο χρόνο απόκρισης. Ειδικά σε περιπτώσεις εφαρμογών που συλλέγουν δεδομένα ώστε να τα χρησιμοποιήσουν για να λάβουν αποφάσεις σε πραγματικό χρόνο, οι χρήστες θα απογοητεύονταν περιμένοντας τα δεδομένα να σταλούν στο cloud και να επιστρέψουν οι απαντήσεις. Παράλληλα, είναι αμφιλεγόμενο το πόσες τέτοιες εφαρμογές μπορεί να υποστηρίξει το σύστημα την ίδια χρονική στιγμή [9].

Εκτός από την πολύ σημαντική μείωση του χρόνου απόκρισης, ξεχωρίζουν και άλλα στοιχεία όπως είναι η αποδοτικότερη αξιοποίηση του εύρους, η ελάττωση του ενεργειακού κόστους, η αποφυγή υπερβολικής κίνησης στο δίκτυο και η αρωγή που προσφέρει στην αποτελεσματική αποφυγή εμφάνισης συμφόρησης.

Κεφάλαιο 2^ο: Μεθοδολογία

Για τον έλεγχο της ευστάθειας του μηχανισμού σε δίκτυα NDN και της αποδοτικής λειτουργίας του, προτιμήθηκε η μέθοδος των πειραμάτων προσομοίωσης. Για την προσομοίωση επιλέχθηκε το περιβάλλον του προσομοιωτή NDN δικτύων ndnSIM [10] (ο οποίος βασίζεται στο προσομοιωτή δικτύων ns-3 [11]). Τα αποτελέσματα καταγράφηκαν σε trace files και για την επεξεργασία τους δημιουργήθηκαν κατάλληλα scripts στη γλώσσα R [12].

2.1 Γενικές Παράμετροι και Μετρικές

- Η τοπολογία δικτύου που επιλέχθηκε για την προσομοίωση των πραγματικών δικτύων είναι η τοπολογία δέντρου. Σε κάθε προσομοίωση ένα υποσύνολο των φύλλων της τοπολογίας δέντρου αποτελεί ζεύγη παραγωγών - καταναλωτών. Το πλήθος των ζευγών δεν είναι σταθερό για όλα τα πειράματα. Πειραματιζόμαστε με τοπολογίες που έχουν 10,20,25 ή 50 ζεύγη.
- Ανάμεσα στη ρίζα του δέντρου και τα φύλλα μεσολαβούν routers. Ρυθμίζουμε όλες τις συνδέσεις των φύλλων με τα routers ώστε να έχουν εύρος ζώνης 100 Mbps, καθυστέρηση 10 ms, μέγιστο επιτρεπόμενο αριθμό πακέτων στην ουρά για μετάδοση ίσο με 10 πακέτα και παράγοντα/βάρος για τη δρομολόγηση ίσο με 1.
- Ρυθμίζουμε όλες τις συνδέσεις των routers με τη ρίζα ώστε να έχουν εύρος ζώνης 10 Mbps, καθυστέρηση 10 ms, μέγιστο επιτρεπόμενο αριθμό πακέτων στην ουρά για μετάδοση ίσο με 10 πακέτα και παράγοντα/βάρος για τη δρομολόγηση ίσο με 1.

Αυτή η δομή αποτελεί το υπόβαθρο για την υλοποίηση της εφαρμογής. Σ' αυτό το σημείο θα πρέπει να τονίσουμε ότι ο μηχανισμός που έχουμε υλοποιήσει δεν εξετάζει την περίπτωση δημιουργίας συμφόρησης στο δίκτυο.

Οι μετρικές που ελέγχονται τελικά, κυρίως για τη σύγκριση μεταξύ των δύο προτεινόμενων σεναρίων, είναι το throughput στη ρίζα της τοπολογίας και το πλήθος των Interests που ικανοποιούνται. Δίνεται, δηλαδή, έμφαση στην κίνηση που αναπτύσσεται στο δίκτυο. Τα στοιχεία αυτά τα εξάγουμε από τα trace files.

2.2 Σενάρια

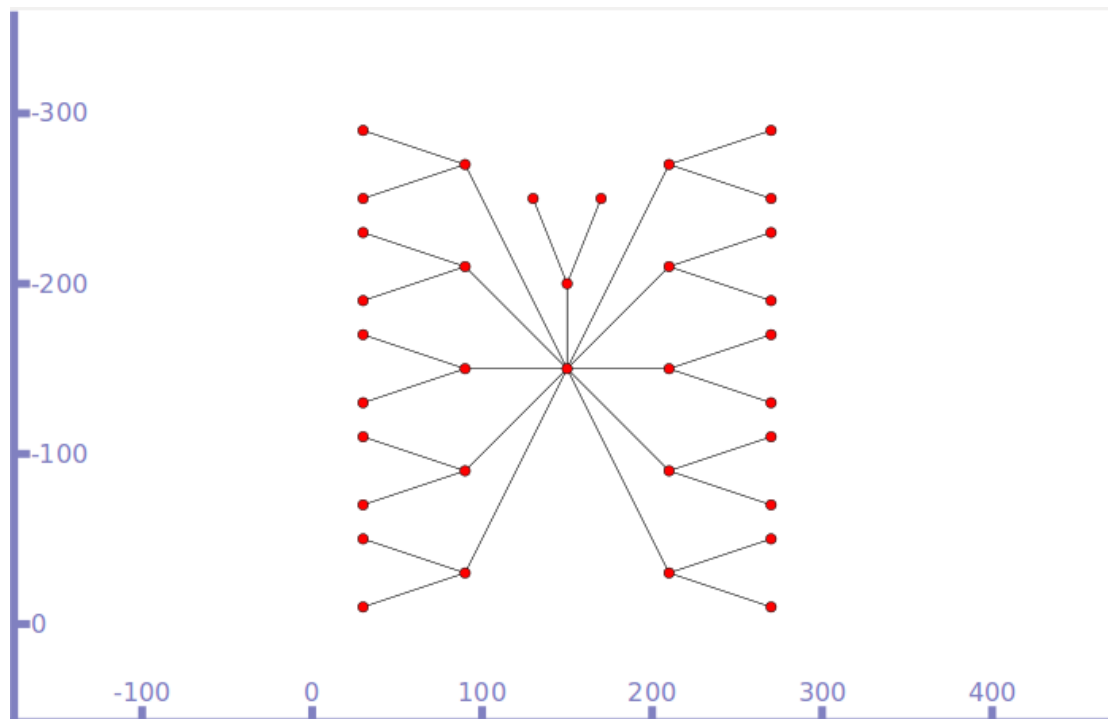
Η εφαρμογή υλοποιήθηκε σε δύο διαφορετικά σενάρια. Στο πρώτο η λειτουργία ακολουθεί τη φιλοσοφία του Cloud Computing. Σύμφωνα με αυτό τα δεδομένα που συλλέγονται στέλνονται σε ένα κεντρικό κόμβο για να τα διαχειριστεί και να τα επεξεργαστεί. Το ρόλο αυτού του κόμβου παίζει η ρίζα της τοπολογίας. Σε αυτό το σενάριο έχουμε μια πιο πολύπλοκη και σύνθετη υλοποίηση, αφού αναγκαστικά πρέπει να γίνονται συχνοί έλεγχοι σχετικά με την απώλεια επιπλέον δεδομένων για αποστολή προτού συνεχίσει ο κώδικας στο επόμενο βήμα.

Το δεύτερο σενάριο βασίζεται στην τεχνική του Edge Computing. Εδώ η επεξεργασία των πακέτων που συλλέχθηκαν γίνεται τοπικά, χωρίς να χρειάζεται να συγκεντρωθούν σε μια κεντρική θέση. Αυτή η λογική συνδυάζει τη δυνατότητα μιας απλοποιημένης υλοποίησης με οφέλη που σχετίζονται με τη μειωμένη κίνηση στο δίκτυο.

2.3 Τοπολογία Δικτύου

Όπως αναφέρθηκε ήδη, έχουμε προτιμήσει την τοπολογία δέντρου για την πραγματοποίηση των προσομοιώσεών μας. Έχουμε δοκιμάσει την εφαρμογή για 10,20,25 και 50 καταναλωτές και άλλους τόσους παραγωγούς.

- 10 καταναλωτές/10 παραγωγοί



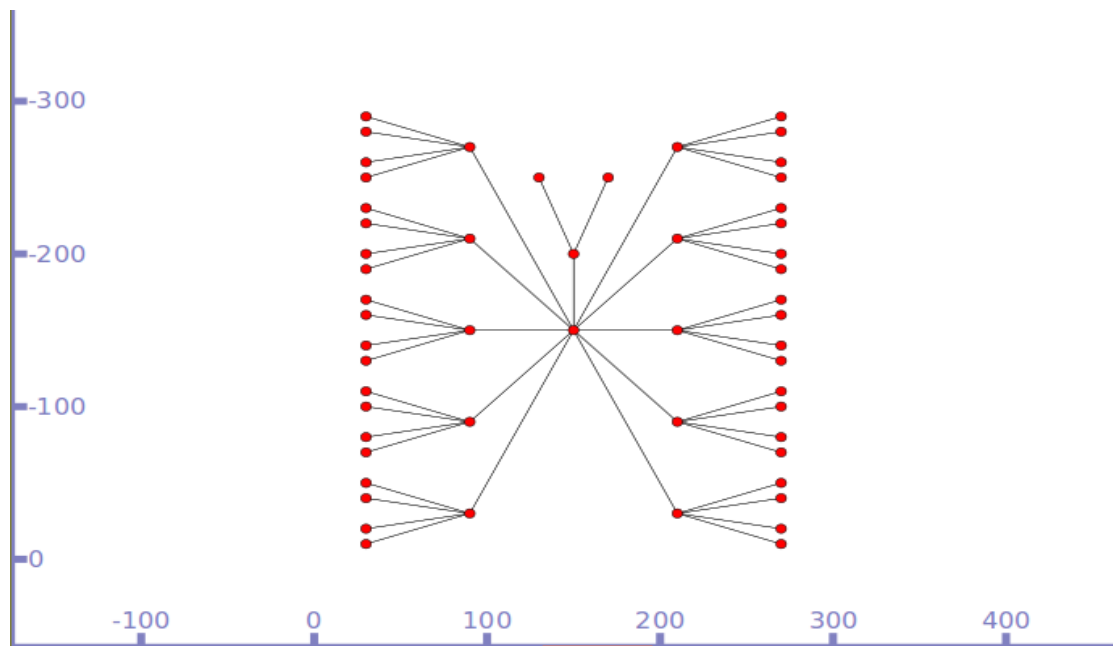
Εικόνα 5: Τοπολογία 10 ζευγών

Στην παραπάνω τοπολογία:

- Τα δέκα αριστερότερα φύλλα αποτελούν τους παραγωγούς των πακέτων δεδομένων.
- Τα δέκα δεξιότερα φύλλα αποτελούν τους καταναλωτές των πακέτων δεδομένων. Αυτοί είναι που αξιολογούν τις πληροφορίες.
- Ο μεσαίος κόμβος, στη θέση (150,150), είναι η ρίζα του δέντρου και αποτελεί τον κόμβο που συγκεντρώνει τις βαθμολογίες και παράγει μια μέση βαθμολογία.
- Τα δύο φύλλα στις θέσεις (130,250) και (170,250) αποτελούν τους νέους καταναλωτές που συνδέονται στο τέλος.
- Όλοι οι ενδιάμεσοι κόμβοι αποτελούν routers.

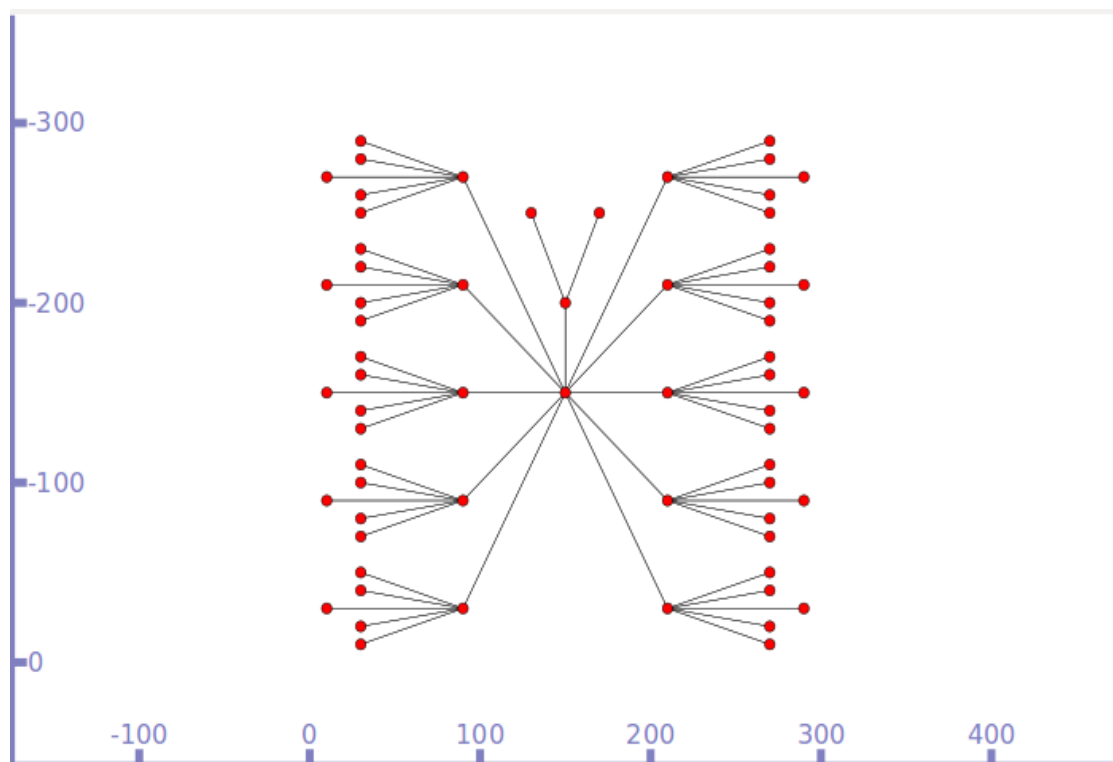
Με όμοιο τρόπο διαμορφώνονται και οι υπόλοιπες τοπολογίες.

- 20 καταναλωτές/20 παραγωγοί



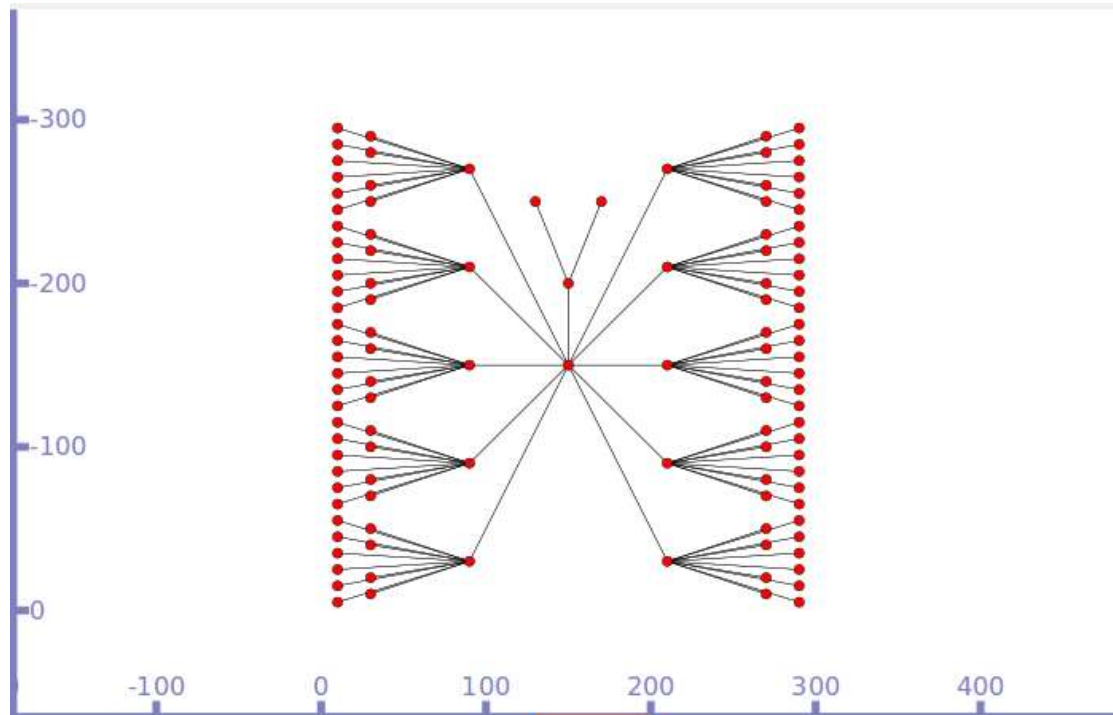
Εικόνα 6: Τοπολογία 20 ζευγών

- 25 καταναλωτές/25 παραγωγοί



Εικόνα 7: Τοπολογία 25 ζευγών

- 50 καταναλωτές/50 παραγωγοί



Εικόνα 8: Τοπολογία 50 ζευγών

Κεφάλαιο 3^ο: Μηχανισμός

Ο μηχανισμός αξιολόγησης της εμπιστοσύνης απαρτίζεται από τέσσερις εφαρμογές που εγκαθίστανται στους κόμβους του δικτύου ως Application Helpers και τρέχουν παράλληλα για την επίτευξη της επιθυμητής λειτουργίας. Αυτές είναι οι ConsumerTrust, ProducerTrust, CustomApp και NewConsumer. Στη συνέχεια του κεφαλαίου αναλύεται ο ρόλος καθεμιάς σε καθένα από τα δύο προτεινόμενα σενάρια. Η τοπολογία δικτύου που επιλέχθηκε για την προσομοίωση των πραγματικών δικτύων είναι η τοπολογία δέντρου.

3.1 1^η Εκδοχή

Στο πρώτο σενάριο που σχεδιάστηκε η λειτουργία του δικτύου θα μπορούσε να παρομοιαστεί με το cloud computing. Πιο αναλυτικά τα βήματα που ακολουθούνται έχουν ως εξής :

- Τα φύλλα της τοπολογίας, εκτός από δύο, χωρίζονται ισομερώς σε καταναλωτές και παραγωγούς. Στους κόμβους αυτούς εγκαθίστανται οι εφαρμογές ConsumerTrust και ProducerTrust αντίστοιχα.
- Στη ConsumerTrust δημιουργείται μια δομή (struck) με μέλη το όνομα του πακέτου (dataName), τον παραγωγό του πακέτου (producer) και τη βαθμολογία που έλαβε (dataRating). Αυτή η δομή χρησιμεύει για την αποθήκευση των στοιχείων των ληφθέντων πακέτων του κάθε consumer και των αντίστοιχων βαθμολογιών.
- Στην πρώτη φάση, όλοι οι consumers ταυτόχρονα στέλνουν από ένα Interest Packet σε κάποιον τυχαίο producer (Συνάρτηση ConsumerTrust::SendInterest). Η εφαρμογή αποθηκεύει προσωρινά σε ένα struck το Interest που έστειλε κάθε κόμβος και το id του κόμβου.
- Τα Interests δρομολογούνται στους producers οι οποίοι μόλις τα λάβουν (Συνάρτηση ProducerTrust::OnInterest) απαντούν στέλνοντας πίσω Data Packets (Συνάρτηση ProducerTrust::SendData).
- Μόλις ληφθούν τα Data Packets (Συνάρτηση ConsumerTrust::OnData) διανέμονται, δηλαδή δρομολογείται το καθένα στο consumer που το ζήτησε.

- Κάθε consumer βαθμολογεί το ληφθέν πακέτο του με μια τυχαία τιμή σε μια κλίμακα από 1 έως 10 (Συνάρτηση `ConsumerTrust::RateData`).
- Τα προηγούμενα επαναλαμβάνονται έως ότου κάθε consumer να λάβει και να αποθηκεύσει 100 πακέτα. Τότε εκτυπώνονται τα δεδομένα που έχει συλλέξει κάθε consumer (Συνάρτηση `ConsumerTrust::Print`).
- Σ' αυτή τη χρονική στιγμή (μετά από πειράματα καταλήξαμε στα 100 seconds) εγκαθίσταται στη ρίζα της τοπολογίας η εφαρμογή `CustomApp`.
- Η `CustomApp` διατηρεί ένα δισδιάστατο πίνακα για την καταχώρηση όλων των βαθμολογήσεων. Κάθε στήλη αντιστοιχεί στις βαθμολογίες που έχουν λάβει τα `Data Packets` του κάθε producer. Οι βαθμολογίες του κάθε consumer μπαίνουν διαδοχικά στις γραμμές της κάθε στήλης. Με άλλα λόγια, η πρώτη στήλη περιέχει όλες τις βαθμολογίες που έχει μαζέψει ο πρώτος producer από όλους τους consumers σε σειρά, η δεύτερη στήλη τις βαθμολογίες του δεύτερου producer κ.ο.κ. Στα επόμενα βήματα εξηγείται πως συμπληρώνονται τα στοιχεία του πίνακα αυτού.
- Ξεκινώντας, η `CustomApp` ζητάει τις βαθμολογίες του πρώτου consumer για τον πρώτο producer (Συνάρτηση `CustomApp::SendInterest`).
- Ο consumer απαντάει στέλνοντας κάθε φορά την επόμενη βαθμολογία που έχει για τον πρώτο producer (Συνάρτηση `ConsumerTrust::OnData`) μέχρι αυτές να τελειώσουν.
- Η κάθε φορά ληφθείσα βαθμολογία αποθηκεύεται στην επόμενη γραμμή της πρώτης στήλης (Συνάρτηση `CustomApp::OnData`).
- Κάθε δευτερόλεπτο ελέγχεται εάν έχει προστεθεί νέα βαθμολογία στην πρώτη στήλη. Εάν διαπιστωθεί πως για τρία συνεχόμενα δευτερόλεπτα δεν έγινε νέα προσθήκη σημαίνει πως μπορεί να προχωρήσει στον επόμενο producer (Συνάρτηση `CustomApp::CheckRows`).
- Όταν έχουν ελεγχθεί όλοι οι producers εμφανίζεται η έως τότε κατάσταση του πίνακα και ο κώδικας προχωράει στον επόμενο consumer (Συνάρτηση `CustomApp::ChangeConsumer`).
- Τα τελευταία πέντε βήματα επαναλαμβάνονται για όλους τους consumers.
- Μόλις τελειώσουν και οι consumers υπολογίζεται ο μέσος όρος των βαθμολογιών για τον κάθε producer (Συνάρτηση `CustomApp::Aggregate`). Έτσι προκύπτει ένα μέτρο της αξιοπιστίας τους.

- Λίγο πριν τη λήξη της προσομοίωσης εγκαθίσταται στα δύο περισευόμενα φύλλα της τοπολογίας η εφαρμογή NewConsumer. Το νόημα αυτών είναι η αναπαράσταση δύο νέων καταναλωτών που δεν έχουν κάποια προηγούμενη σχέση με το δίκτυο αλλά τη στιγμή που συνδέονται θα έχουν στη διάθεσή τους ένα κριτήριο για το αν θα πρέπει να εμπιστευτούν κάποιον άγνωστο γι' αυτούς παραγωγό.
- Διαδοχικά οι δύο νέοι consumers διαλέγουν έναν τυχαίο producer από τον οποίο ζητούν κάποιο Data Packet (Συνάρτηση NewConsumer::SendInterest).
- Μόλις λάβουν το πακέτο ζητούν από τη ρίζα να τους στείλει τη μέση βαθμολογία του ίδιου producer (Συναρτήσεις NewConsumer::SendInterest , CustomApp::OnInterest). Είναι τότε σε θέση να κρίνουν πόσο έμπιστος είναι ο παραγωγός που επέλεξαν.

3.2 2^η Εκδοχή

Στο δεύτερο σενάριο που μελετάται έχουμε αλλάξει εν μέρη τη λογική ώστε να πλησιάζει τη φιλοσοφία του edge computing. Πιο αναλυτικά τα βήματα που ακολουθούνται έχουν ως εξής:

- Τα φύλλα της τοπολογίας, εκτός από δύο, χωρίζονται ισομερώς σε καταναλωτές και παραγωγούς. Στους κόμβους αυτούς εγκαθίστανται οι εφαρμογές ConsumerTrust και ProducerTrust αντίστοιχα.
- Στη ConsumerTrust δημιουργείται μια δομή (struck) με μέλη το όνομα του πακέτου (dataName), τον παραγωγό του πακέτου (producer) και τη βαθμολογία που έλαβε (dataRating). Αυτή η δομή χρησιμεύει για την αποθήκευση των στοιχείων των ληφθέντων πακέτων του κάθε consumer και των αντίστοιχων βαθμολογιών.
- Στην πρώτη φάση, όλοι οι consumers ταυτόχρονα στέλνουν από ένα Interest Packet σε κάποιον τυχαίο producer (Συνάρτηση ConsumerTrust::SendInterest). Η εφαρμογή αποθηκεύει προσωρινά σε ένα struck το Interest που έστειλε κάθε κόμβος και το id του κόμβου.

- Τα Interests δρομολογούνται στους producers οι οποίοι μόλις τα λάβουν (Συνάρτηση ProducerTrust::OnInterest) απαντούν στέλνοντας πίσω Data Packets (Συνάρτηση ProducerTrust::SendData).
- Μόλις ληφθούν τα Data Packets (Συνάρτηση ConsumerTrust::OnData) διανέμονται, δηλαδή δρομολογείται το καθένα στο consumer που το ζήτησε.
- Κάθε consumer βαθμολογεί το ληφθέν πακέτο του με μια τυχαία τιμή σε μια κλίμακα από 1 έως 10 (Συνάρτηση ConsumerTrust::RateData).
- Τα προηγούμενα επαναλαμβάνονται έως ότου κάθε consumer να λάβει και να αποθηκεύσει 100 πακέτα.
- Έπειτα κάθε consumer υπολογίζει ένα μέσο όρο από τις βαθμολογίες που έχει βάλει για κάθε producer καθώς και το πλήθος των βαθμολογιών που έχει βάλει σε κάθε producer (Συνάρτηση ConsumerTrust::Aggregate).
- Σ' αυτή τη χρονική στιγμή (μετά από πειράματα καταλήξαμε στα 100 seconds) εγκαθίσταται στη ρίζα της τοπολογίας η εφαρμογή CustomApp.
- Η CustomApp διατηρεί δύο διδιάστατους πίνακες: έναν για την καταχώρηση όλων των υπολογισμένων μέσων όρων ($Ratings[N][N]$) και έναν για την καταχώρηση όλων των υπολογισμένων πληθών ($Number[N][N]$). Με άλλα λόγια, η πρώτη στήλη του πρώτου πίνακα περιέχει όλους τους μέσους όρους που έχουν υπολογίσει όλοι οι consumers για τον πρώτο producer, η δεύτερη στήλη όλους τους μέσους όρους που έχουν υπολογίσει όλοι οι consumers για τον δεύτερο producer κ.ο.κ. Ομοίως, η πρώτη στήλη του δεύτερου πίνακα περιέχει όλα τα πλήθη βαθμολογιών που έβαλαν όλοι οι consumers για τον πρώτο producer, η δεύτερη στήλη όλα τα πλήθη βαθμολογιών που έβαλαν όλοι οι consumers για τον δεύτερο producer κ.ο.κ.
- Ξεκινώντας, η CustomApp ζητάει το μέσο όρο και το πλήθος βαθμολογιών του πρώτου consumer για τον πρώτο producer (Συνάρτηση CustomApp::SendInterest).
- Ο consumer απαντάει στέλνοντας τα υπολογισμένα μεγέθη για τον πρώτο producer (Συνάρτηση ConsumerTrust::OnInterest).
- Η CustomApp αποθηκεύει τα δεδομένα στους πίνακες και προχωράει στον επόμενο producer (Συνάρτηση CustomApp::OnData).
- Τα τελευταία τρία βήματα επαναλαμβάνονται για όλους τους consumers.

- Μόλις τελειώσουν οι consumers υπολογίζεται ένας σταθμισμένος μέσος όρος για κάθε i-οστό producer ως εξής :

$$Average[i] = \frac{Ratings[0][i] \times Number[0][i] + Ratings[1][i] \times Number[1][i] + \dots + Ratings[N][i] \times Number[N][i]}{Number[0][i] + Number[1][i] + \dots + Number[N][i]}$$

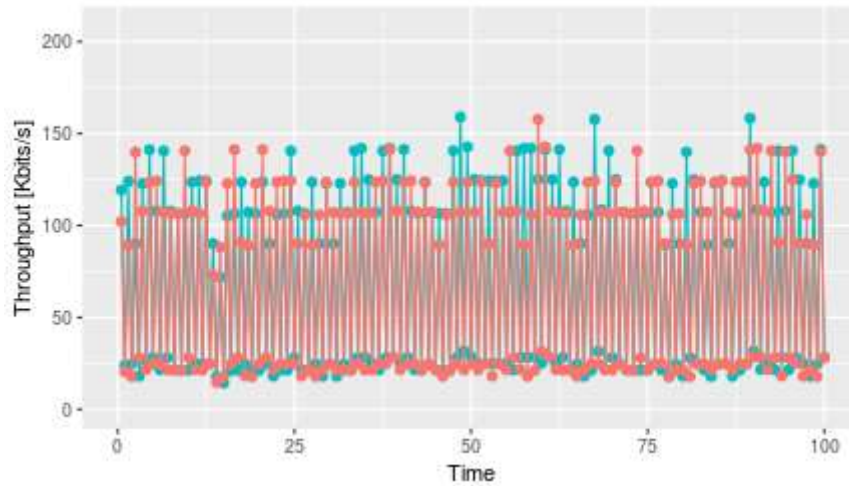
Αυτός αποτελεί ένα μέτρο της αξιοπιστίας τους.

- Λίγο πριν τη λήξη της προσομοίωσης εγκαθίσταται στα δύο περισευόμενα φύλλα της τοπολογίας η εφαρμογή NewConsumer. Το νόημα αυτών είναι η αναπαράσταση δύο νέων καταναλωτών που δεν έχουν κάποια προηγούμενη σχέση με το δίκτυο αλλά τη στιγμή που συνδέονται θα έχουν στη διάθεσή τους ένα κριτήριο για το αν θα πρέπει να εμπιστευτούν κάποιον άγνωστο γι' αυτούς παραγωγό.
- Διαδοχικά οι δύο νέοι consumers διαλέγουν έναν τυχαίο producer από τον οποίο ζητούν κάποιο Data Packet (Συνάρτηση NewConsumer::SendInterest).
- Μόλις λάβουν το πακέτο ζητούν από τη ρίζα να τους στείλει το σταθμισμένο μέσο όρο που έχει προκύψει για τον ίδιο producer (Συναρτήσεις NewConsumer::SendInterest , CustomApp::OnInterest). Είναι τότε σε θέση να κρίνουν πόσο έμπιστος είναι ο παραγωγός που επέλεξαν.

4.1 Σύγκριση της Απόδοσης (Throughput)

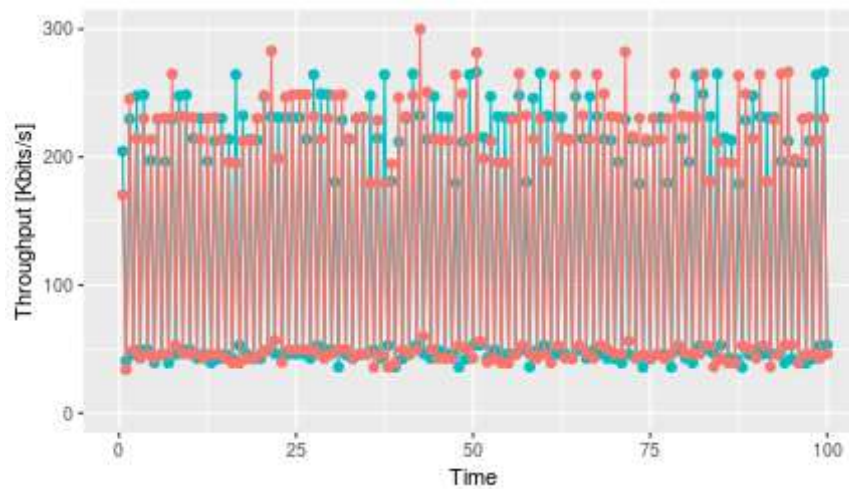
4.1.1 Απόδοση τα πρώτα 100 secs της προσομοίωσης

- **10 consumers Throughput**



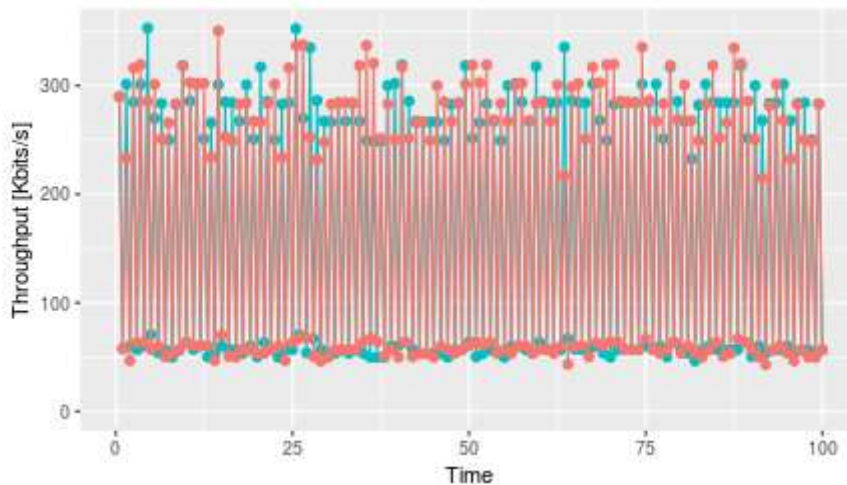
Εικόνα 9: Σύγκριση των Throughput στη ρίζα για 10 καταναλωτές τα 100 πρώτα δευτερόλεπτα

- **20 consumers Throughput**



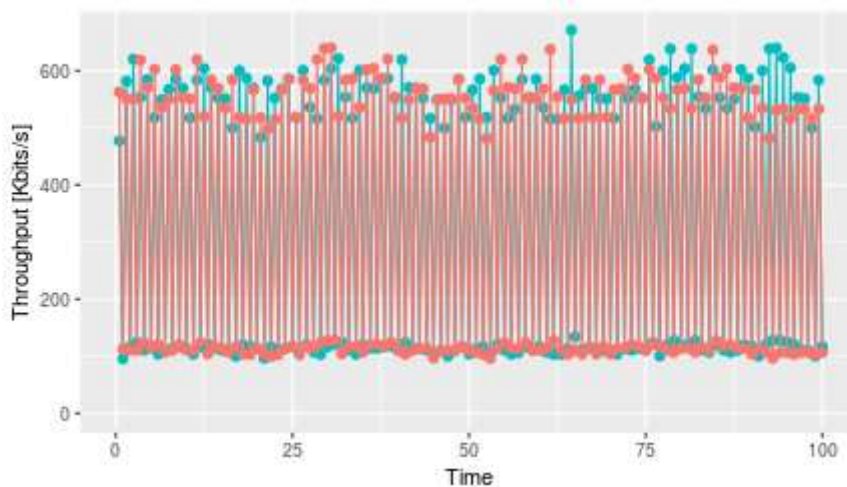
Εικόνα 10: Σύγκριση των Throughput στη ρίζα για 20 καταναλωτές τα 100 πρώτα δευτερόλεπτα

- **25 consumers Throughput**



Εικόνα 11: Σύγκριση των Throughput στη ρίζα για 25 καταναλωτές τα 100 πρώτα δευτερόλεπτα

- **50 consumers Throughput**

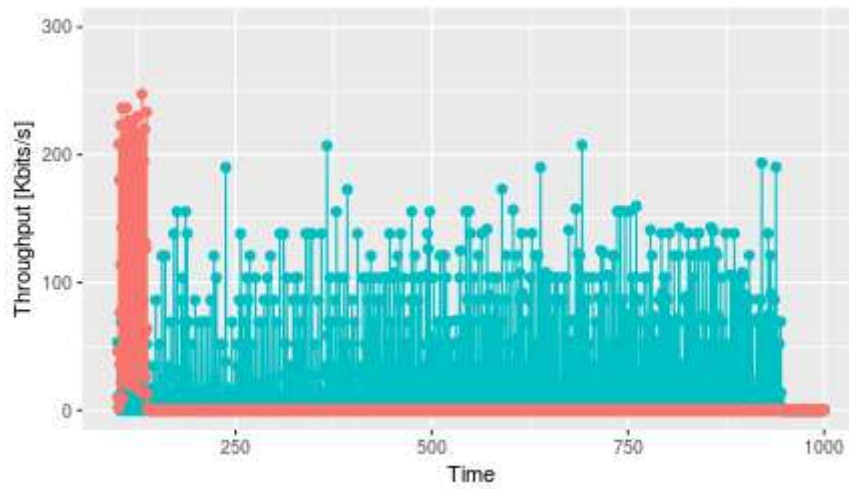


Εικόνα 12: Σύγκριση των Throughput στη ρίζα για 50 καταναλωτές τα 100 πρώτα δευτερόλεπτα

Και στα δύο σενάρια, η αύξηση των καταναλωτών και των παραγωγών συνεπάγεται την αύξηση της απόδοσης της ρίζας. Αυτό είναι λογικό εφόσον τη στιγμή που στην αρχή στέλνονταν 10 Interests και γυρνούσαν 10 Data Packets, έπειτα στέλνονται και γυρνάνε 20,25 ή 50 Interests και Data Packets αντίστοιχα. Αυτό που είναι αξιόλογο όμως, είναι πως αν παρατηρήσει κανείς μπορεί να διαπιστώσει πως η αύξηση της απόδοσης ακολουθεί σχεδόν αναλογικά την αύξηση των κόμβων.

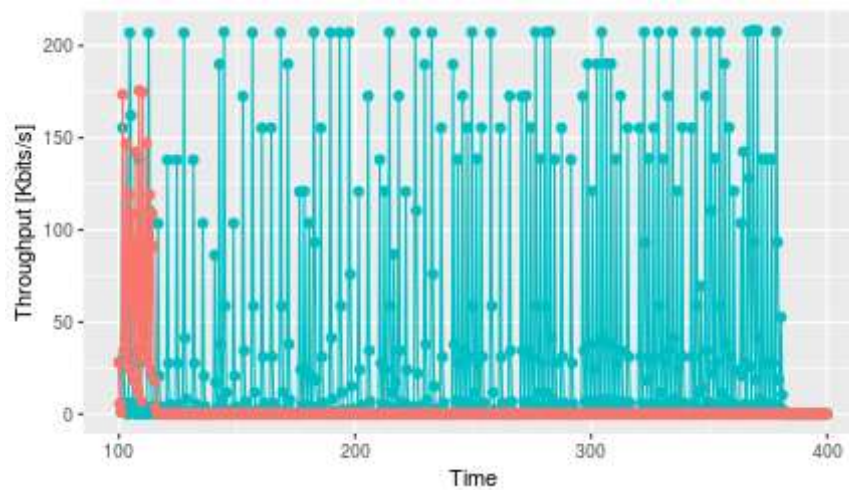
4.1.2 Απόδοση μετά τα πρώτα 100 secs της προσομοίωσης

- **10 consumers Throughput**



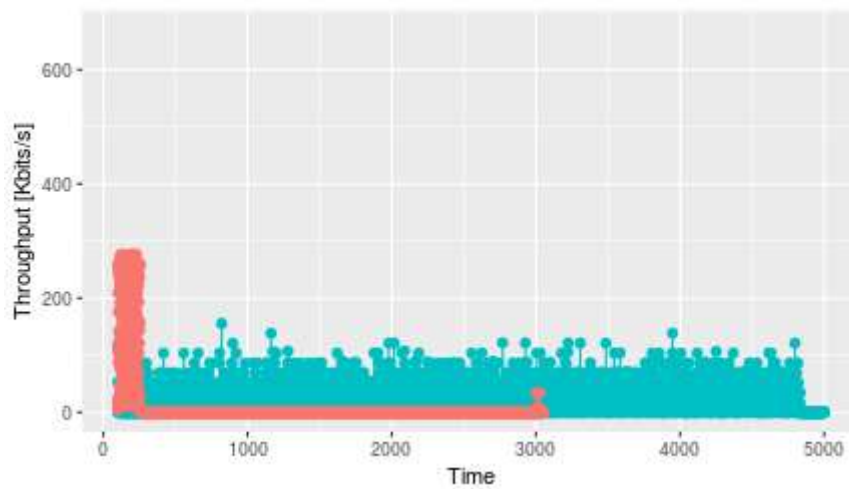
Εικόνα 13: Σύγκριση Throughput στη ρίζα για 10 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα

- **20 consumers Throughput**



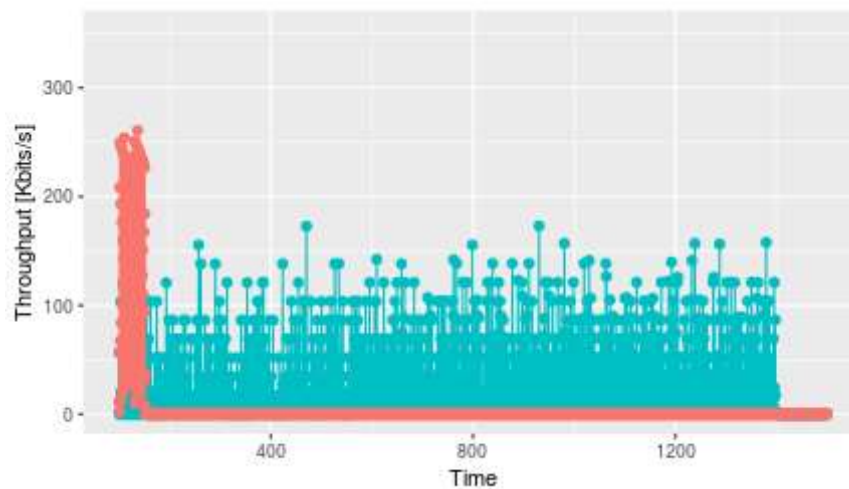
Εικόνα 14: Σύγκριση Throughput στη ρίζα για 20 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα

- **25 consumers Throughput**



Εικόνα 15: Σύγκριση Throughput στη ρίζα για 25 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα

- **50 consumers Throughput**



Εικόνα 16: Σύγκριση Throughput στη ρίζα για 50 καταναλωτές μετά τα 100 πρώτα δευτερόλεπτα

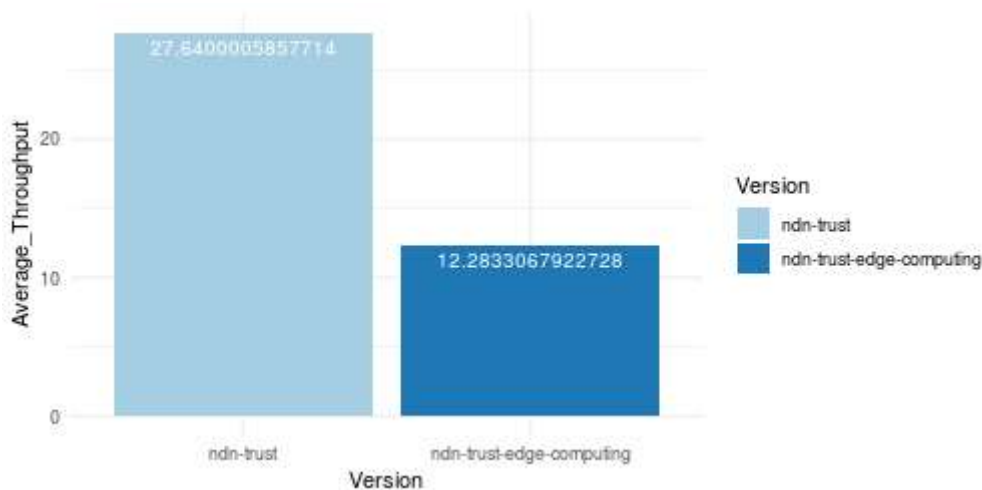
Όσον αφορά την πρώτη εκδοχή, όσο αυξάνεται ο αριθμός των ζευγών καταναλωτών – παραγωγών, τόσο ελαττώνεται το throughput. Αυτό μπορεί να εξηγηθεί με βάση την θεωρία των πιθανοτήτων. Αναλυτικότερα, όσο μεγαλώνει το σύνολο των παραγωγών, τόσο μειώνεται η πιθανότητα ένας συγκεκριμένος παραγωγός να εκλεχτεί τυχαία από

έναν ορισμένο καταναλωτή. Επομένως, το μέγεθος των δεδομένων που έχουν συγκεντρώσει για κάθε παραγωγό οι καταναλωτές είναι σχετικά μικρότερο. Βέβαια, η διαφορά αυτή δεν είναι ιδιαίτερα σημαντική. Φυσικά, λόγω της τυχαιότητας της επιλογής των παραγωγών η μορφή των σχεδιαγραμμάτων και οι αποστάσεις ανάμεσά τους δεν είναι ίδιες σε κάθε εκτέλεση.

Από την άλλη μεριά, στη δεύτερη εκδοχή της εφαρμογής, η εικόνα είναι εντελώς διαφορετική. Χάρη στο edge computing, η επεξεργασία των συλλεγόμενων δεδομένων γίνεται στους καταναλωτές οπότε δε χρειάζεται να στείλουν όλες τις λεπτομερείς πληροφορίες στη ρίζα. Αρκούνται, λοιπόν, στο να στείλουν ένα πακέτο για κάθε παραγωγό με τη μέση βαθμολογία του και το πλήθος των βαθμολογήσεων του. Η αποστολή αυτών των δεδομένων χρειάζεται ένα σύντομο χρονικό διάστημα για να ολοκληρωθεί (περίπου ίσο με 160 δευτερόλεπτα στη χειρότερη περίπτωση). Αυτό έχει σαν αποτέλεσμα, όπως φαίνεται και στα γραφήματα, το throughput του δικτύου σε αυτό το στάδιο να γίνεται έπειτα σχεδόν μηδαμινό.

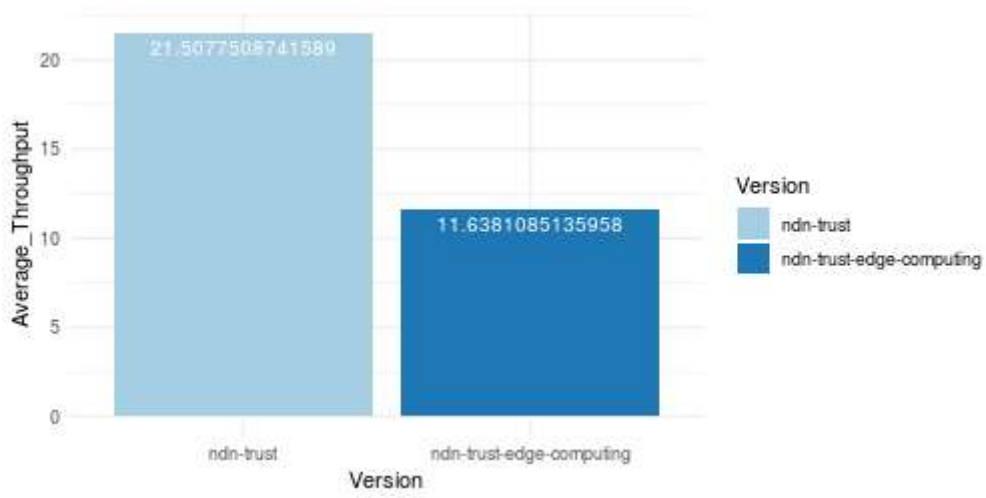
4.2 Σύγκριση μέσης Απόδοσης

- **10 consumers Throughput**



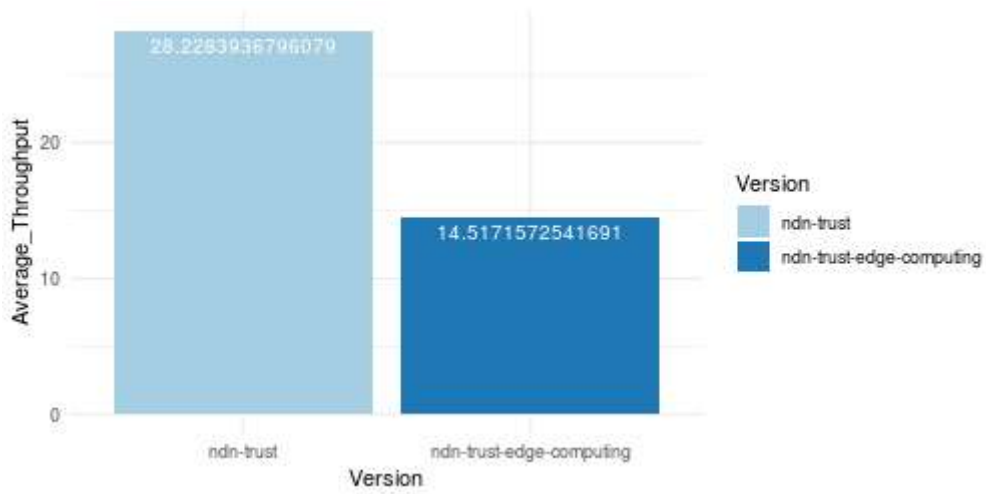
Εικόνα 17: Σύγκριση μέσου Throughput στη ρίζα για 10 καταναλωτές

- **20 consumers Throughput**



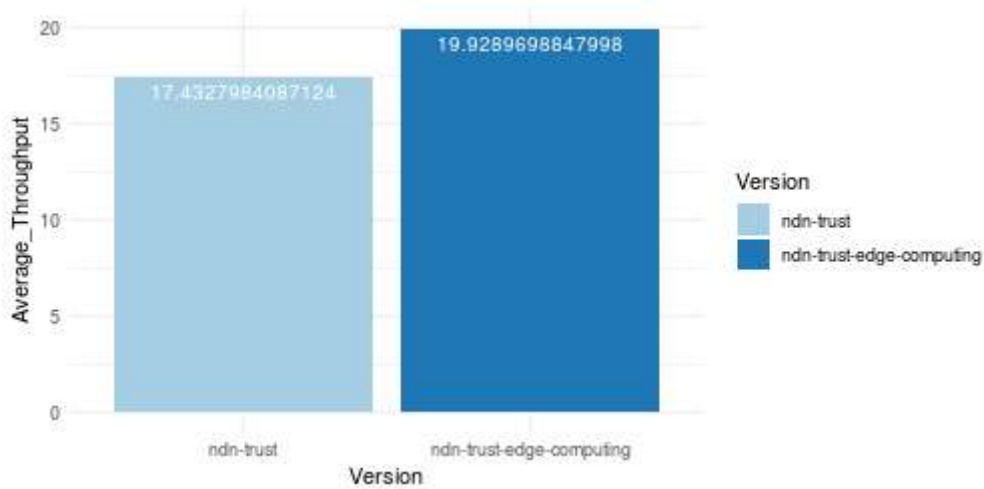
Εικόνα 18: Σύγκριση μέσου Throughput στη ρίζα για 20 καταναλωτές

- **25 consumers Throughput**



Εικόνα 19: Σύγκριση μέσου Throughput στη ρίζα για 25 καταναλωτές

- **50 consumers Throughput**



Εικόνα 20: Σύγκριση μέσου Throughput στη ρίζα για 50 καταναλωτές

Όσο αυξάνεται το πλήθος των καταναλωτών, η διαφορά ανάμεσα στα δύο σενάρια αμβλύνεται. Αυτό συμβαίνει επειδή υπολογίζουμε τη μέση απόδοση ανεξάρτητα από τον αριθμό των πακέτων και το χρονικό διάστημα μέσα στο οποίο διακινούνται. Αν και οι τιμές της μέσης απόδοσης της πρώτης εκδοχής ακολουθούν μια καθοδική πορεία, δε φαίνεται να συμβαίνει το ίδιο στην δεύτερη περίπτωση. Ωστόσο, σημαντικός είναι και πάλι ο παράγοντας της τυχαιότητας σε κάθε εκτέλεση.

5.1 Γενικά συμπεράσματα

Έχουμε σχεδιάσει μια εφαρμογή αξιολόγησης της εμπιστοσύνης και αποδειξίει ότι μπορεί να υποστηριχθεί από ένα NDN δίκτυο και να λειτουργήσει σ' αυτό παράγοντας ενθαρρυντικά αποτελέσματα μέσα από κατάλληλες προσομοιώσεις. Επίσης, είναι μια καλή βάση για τη δημιουργία μιας πιο πολύπλοκης εφαρμογής αξιολόγησης, εφόσον εξασφαλίζει ένα κριτήριο της πιστότητας των παραγωγών πληροφοριών και παρέχει μια εκτίμηση σε χρήστες που είναι άγνωστοι με το δίκτυο.

Εστιάζοντας στα εξεταζόμενα σενάρια συμπεραίνει κανείς εύκολα από τα προηγούμενα πως η τεχνική του Edge Computing υπερτερεί έναντι του Cloud Computing καθώς αποφορτίζει αισθητά το δίκτυο.

5.2 Μελλοντικές Κατευθύνσεις

Καθώς η παρούσα εργασία αποτελεί το αρχικό στάδιο στο σχεδιασμό ενός πρωτοκόλλου εμπιστοσύνης, άμεσος στόχος είναι η δημιουργία ενός ολοκληρωμένου πρωτοκόλλου. Μερικές πιθανές μελλοντικές κατευθύνσεις είναι:

- η άτακτη συλλογή των αρχικών δεδομένων από τους καταναλωτές, ακολουθώντας μια κατανομή, όπως η zipf, ώστε να προσομοιώνονται πραγματικές συνθήκες
- η εκτέλεση προσομοιώσεων όπου προκύπτει συμφόρηση στο δίκτυο
- η λειτουργία των καταναλωτών χωρίς περιορισμό στα πακέτα που συλλέγει ο καθένας και χωρίς τον μεταξύ τους συγχρονισμό, συνεχής αναδιαμόρφωση της βαθμολογίας των παραγωγών, μη προγραμματισμένη σύνδεση και αποσύνδεση νέων καταναλωτών
- ο πειραματισμός σε πραγματικές τοπολογίες

Επίσης, σημαντική προτεραιότητα τίθεται η προσαρμογή του μηχανισμού ώστε να ενσωματωθεί στη διαστημική τεχνολογία. Συγκεκριμένα, σκοπεύουμε να εξετάσουμε

σενάρια όπου ένα δίκτυο αισθητήρων σε μια διαστημική βάση συλλέγει δεδομένα από δορυφόρους τα οποία μεταφέρονται με κόμβους/οχήματα όπου είναι εγκατεστημένα κατάλληλα πρωτόκολλα (π.χ. Delay-Tolerant Networking) [13] σε μια επίγεια βάση. Τότε οι τελικοί κόμβοι στη γη θα μπορούν να αξιολογούν τα ληφθέντα πακέτα με βάση κριτήρια, όπως η καθυστέρηση, για να επιλέγουν ανάλογα το όχημα που θεωρούν ότι θα εξυπηρετήσει ταχύτερα το αίτημά τους να ανακτήσουν διαστημικά δεδομένα.

Επιπροσθέτως, μια σημαντική κατεύθυνση είναι η εφαρμογή του μηχανισμού σε αρχιτεκτονικές και σενάρια που αξιοποιούν πρωτόκολλα διαστημικής τεχνολογίας, όπως παράδειγμα αυτές που περιγράφονται στις αναφορές [14],[15]. Τέλος, μας ενδιαφέρει να ερευνήσουμε την αξιοποίηση του μηχανισμού σε πρωτόκολλα δρομολόγησης [16], [17].

Βιβλιογραφία

- [1] «Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Crowley, P., Papadopoulos, C., ... & Zhang, B. (2014). Named data networking. *ACM SIGCOMM Computer Communication Review*, 44(3), 66-73.».
- [2] «Katsaros, K. V., Chai, W. K., Wang, N., Pavlou, G., Bontius, H., & Paolone, M. (2014). Information-centric networking for machine-to-machine data delivery: a case study in smart grid applications. *IEEE Network*, 28(3), 58-64.».
- [3] «Ahlgren, B., Dannewitz, C., Imbrenda, C., Kutscher, D., & Ohlman, B. (2012). A survey of information-centric networking. *IEEE Communications Magazine*, 50(7), 26-36.».
- [4] «Hassan, S., Elbreiki, W., Firdhous, M., & Habbal, A. M. M. (2015). End-to-End Networks Vs Named Data Network: A Critical Evaluation. *Jurnal Teknologi*, 72(5).».
- [5] «Psaras, I., Chai, W. K., & Pavlou, G. (2012, August). Probabilistic in-network caching for information-centric networks. In *Proceedings of the second edition of the ICN workshop on Information-centric networking* (pp. 55-60). ACM.».
- [6] «Yu, Y., Afanasyev, A., Clark, D., Jacobson, V., & Zhang, L. (2015, September). Schematizing trust in named data networking. In *Proceedings of the 2nd ACM Conference on Information-Centric Networking* (pp. 177-186). ACM.».
- [7] «Gasti, P., & Tsudik, G. (2018, June). Content-Centric and Named-Data Networking Security: The Good, The Bad and The Rest. In *2018 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)* (pp. 1-6). IEEE.».
- [8] «JoSEP, A. D., Katz, R., KonWinSKI, A., Gunho, L. E. E., PAttERSON, D., & RABKIn, A. (2010). A view of cloud computing. *Communications of the ACM*, 53(4).».
- [9] «Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3(5), 637-646.».
- [10] «<https://ndnsim.net/current/>,» [Ηλεκτρονικό].
- [11] «<https://www.nsnam.org/>,» [Ηλεκτρονικό].
- [12] «<https://www.r-project.org/>,» [Ηλεκτρονικό].
- [13] «S. Burleigh, V. Cerf, J. Crowcroft and V. Tsaoussidis, "Space for Internet and Internet for Space", *Ad Hoc Networks, Special Issue on New Research Challenges in Mobile, Opportunistic & Delay-Tolerant Networks*, Elsevier, doi:10.1016/j.adhoc.2014.06.005.».
- [14] «L. Mamas, A. Papadopoulou and V. Tsaoussidis, "Exploiting Communication Opportunities in Disrupted Network Environments," In: *Proceedings of the 13th international conference on Wired/Wireless Internet Communication, Malaga, Spain, 2015*».

- [15] «Sarros, C. A., Diamantopoulos, S., Rene, S., Psaras, I., Lertsinsrubtavee, A., Molina-Jimenez, C., ... & Crowcroft, J. (2018). Connecting the edges: A universal, mobile-centric, and opportunistic communications architecture. *IEEE Communications Magazine*,».
- [16] «G. Araniti, N. Bezirgiannidis, E. Birrane, I. Bisio, S. Burleigh, C. Caini, M. Feldmann, M. Marchese, J. Segui, and K. Suzuki, "Contact graph routing in DTN space networks: overview, enhancements and performance," *Communications Magazine, IEEE* , vol.53, n».
- [17] «Mendes, P., Sofia, R. C., Soares, J., Tsaoussidis, V., Diamantopoulos, S., & Sarros, C. A. (2018, September). Information-centric routing for opportunistic wireless networks. In *Proceedings of the 5th ACM Conference on Information-Centric Networking* (pp.».».