ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΨΗΦΙΑΚΕΣ ΕΠΙΚΟΙΝΩΝΙΕΣ ΚΑΙ ΔΙΚΤΥΑ

ΚΑΤΕΥΘΥΝΣΗ: ΨΗΦΙΑΚΕΣ ΕΠΙΚΟΙΝΩΝΙΕΣ ΚΑΙ ΔΙΚΤΥΑ

Διπλωματική Εργασία

Δίκτυα Καθοριζόμενα Από Λογισμικό

Κατσάρας Δημήτρης

Επιβλέποντες:Αρίστη Γαλάνη

Αθήνα

2018

Διπλωματική Εργασία

Software Defined Networks

Κατσάρας Δημήτρης Α.Μ: ΜΕ1551

Επιβλέπουσα Καθηγήτρια: Αρίστη Γαλάνη

Περίληψη

Η παρούσα διπλωματική εργασία έχει σαν σκοπό να περιγράψει τα Δίκτυα Καθοριζόμενα από το Λογισμικό (Software Defined Networks), τον τρόπο λειτουργίας τους και τα μέρη του δικτύου που επωφελούνται από μια SDN μετάβαση.Αρχικά γίνεται αναφορά των προκλήσεων και των προβλημάτων που αντιμετωπίζει η τωρινή αρχιτεκτονική αλλά και των λύσεων που μπορεί να προσφέρει η μετάβαση σε SDN δίκτυα. Στη συνέχεια γίνεται λεπτομερής αναφορά στο OpenFlow protocol το οποίο είναι από τα πιο σημαντικά εργαλεία επικοινωνίας σε ένα SDN περιβάλλον.Στο 3^{ο} κεφάλαιο δίνεται έμφαση στα οφέλη που έχουμε από την χρήση του SDN στα κινητά και τα ασύρματα δίκτυα.Στο 4^{ο} κεφάλαιο παρατίθεται μία τεχνική η οποία εφαρμόζεται στο πεδίο των δεδομένων (Data Plane) και είναι εξαιρετικά χρήσιμη για διάφορες δικτυακές εφαρμογές.Στο 5^{ο} κεφάλαιο γίνεται προσομοίωση τριών διαφορετικών περιπτώσεων χρήσης ενός SDN δικτύου, χρησιμοποιώντας σαν εργαλείο προσομοίωσης το Mininet σε συνδυασμό με το Virtual Box.Τέλος στο 6^{ο} κεφάλαιο γίνεται η αναφορά των συμπερασμάτων όσον αφορά τα SDN δίκτυα αλλά και των προκλήσεων που πρέπει να ξεπεράσουν .

Abstract

This thesis aims to describe Software Defined Networks their mode of operation plus the network parts that benefit from SDN transition. Initially, we refer to the current challenges and problems that the present network architecture is facing, also we outline the solutions which can be provided by using the SDN architecture. Then a detailed reference to OpenFlow protocol is being made, OpenFlow protocol is one of the most important communication tools in an SDN environment. In the third chapter we emphasize to the benefits we have cause of the use of SDN in Mobile and Wireless networks. Then on chapter four a technique is represented which is applied to the network Data Plane of network devices and is extremely useful for various SDN network applications.In chapter five we simulate three different use cases, using as a simulation tool the Mininet environment in combination with Virtual Box. Finally in chapter six we report the conclusion regarding SDN networks in addition we sumurize some challenges that SDN networks should overcome.

Ευχαριστίες

Η παρούσα διπλωματική αφιερώνεται στην οικογένεια μου, τους φίλους μου και στον Τέλη που μας άφησε νωρίς. Θα ήθελα να ευχαριστήσω θερμά την κυρία Αρίστη Γαλάνη χωρίς την βοήθεια της οποίας θα ήταν αδύνατη η παράδοση της διπλωματικής.

Περιεχόμενα

Κατάλογος Εικόνων

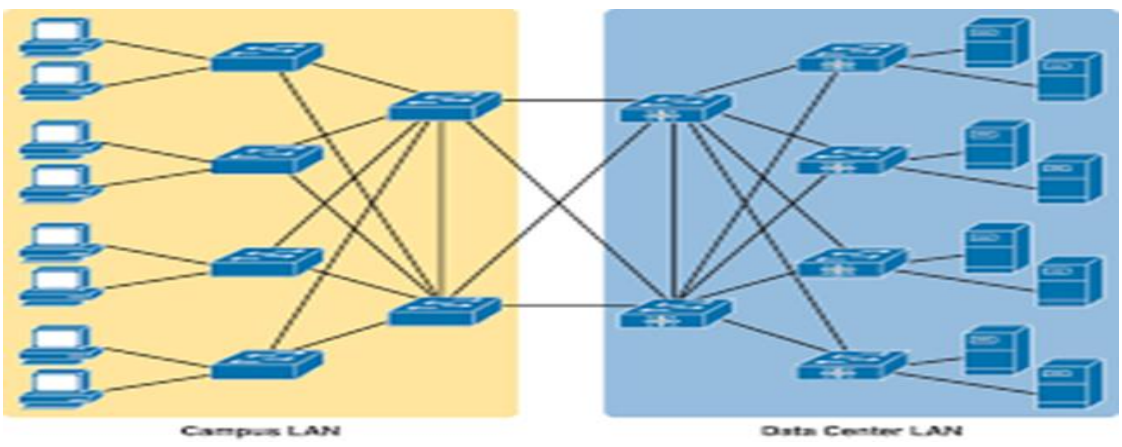
Κατάλογος Πινάκων

Κεφάλαιο 1

1.1Σημερινή Αρχιτεκτονική Δικτύων

Τα δίκτυα υπολογιστών διαδραματίζουν ένα κρίσιμο ρόλο στη σημερινή κοινωνία. Οι υπηρεσίες που προσφέρει το διαδίκτυο όπως είναι για παράδειγμα, οι μηχανές αναζήτησης, τα μέσα κοινωνικής δικτύωσης καθώς και υπηρεσίες ηλεκτρονικού εμπορίου φιλοξενούνται σε κέντρα δεδομένων (data centers). Εκεί εκατοντάδες υπολογιστές συνδέονται με μεγάλα δίκτυα δεδομένων. Τα κέντρα δεδομένων διασυνδέονται μεταξύ τους με δίκτυα ευρείας περιοχής (Wide Area Networks) που καλύπτουν ολόκληρο τον πλανήτη. Οι τελικοί χρήστες χρησιμοποιώντας τους προσωπικούς τους υπολογιστές, τα κινητά τους τηλέφωνα αλλά και άλλες συσκευές επιζητούν πρόσβαση στις υπηρεσίες που προσφέρει το διαδίκτυο μέσω διαφορετικών τεχνολογιών. Για παράδειγμα, οι ενσύρματοι υπολογιστές χρησιμοποιούν τις υπηρεσίες του πρωτοκόλλου Ethernet **[18]** για μεταφορά δεδομένων, οι ασύρματοι υπολογιστές και οι χρήστες κινητών τηλεφώνων χρησιμοποιούν τα πρότυπα ασύρματης δικτύωσης **[19]** για πρόσβαση. Επιπλέον, υπάρχει και η κατηγορία χρηστών που έχει πρόσβαση μέσω ενός κυψελωτού δικτύου. Η διαχείριση αυτών των δικτύων με σκοπό την παροχή γρήγορων, αξιόπιστων και ασφαλών υπηρεσιών, αποτελεί διαχρονικό πρόβλημα.

Τα τωρινά δίκτυα υπολογιστών αποτελούνται από συσκευές (εικόνα 1) όπως οι δρομολογητές (routers), οι μεταγωγείς (switches), τα firewalls (τείχος προστασίας), οι διακομηστές (servers) και ο λόγος χρησιμοποίησης τους είναι η πραγματοποίηση συγκεκριμένων λειτουργιών σε ένα δίκτυο. Για παράδειγμα, ο router είναι υπεύθυνος για την δρομολόγηση της κίνησης μεταξύ δύο τοπικών δικτύων (LANs), το switch το οποίο αντικατέστησε το hub (συσκευή που χρησιμοποιήθηκε αρχικά για την σύνδεση των τερματικών συσκευών σε ένα τοπικό δίκτυο) είναι η συσκευή που χρησιμοποιείται κατά κόρον πλέον για την σύνδεση των τερματικών ενός LAN και οι υπηρεσίες που προσφέρει είναι η υψηλή απόδοση του τοπικού δικτύου και η προώθηση της κίνησης εντός του. Επίσης, τα firewalls (χρησιμοποιούνται είτε σαν συσκευές, είτε υλοποιούνται μέσω software) ορίζουν το είδος της κίνησης (traffic) που επιτρέπεται να περάσει σε ένα δίκτυο. Τέλος, οι Servers (είτε πρόκειται για φυσική συσκευή υλικού, είτε για εικονικές μηχανές που λειτουργούν πάνω από ένα Server) είναι το μέρος όπου βρίσκεται μία ποικιλία εφαρμογών και υπηρεσιών και στις οποίες οι τελικοί χρήστες και οι δικτυακές συσκευές επιθυμούν να έχουν πρόσβαση.



Εικόνα 1: Τοπολογία ενός εταιρικού δικτύου.

Για την επιτυχή μεταφορά της κίνησης, είτε πρόκειται για ένα τοπικό LAN, είτε μεταξύ διαφορετικών LANs, οι διαχειριστές του εκάστοτε δικτύου πρέπει να διαμορφώσουν την κάθε δικτυακή συσκευή ξεχωριστά μία προς μία (συνήθως χειροκίνητα). Στη συνέχεια, οι δικτυακές συσκευές επεξεργάζονται συγκεκριμένες κεφαλίδες ενός πακέτου(π.χ IP διεύθυνση προορισμού, φυσική διεύθυνση προορισμού κ.α) και πραγματοποιούν μία δράση (είτε προωθούν το πακέτο, είτε το απορρίπτουν). Ο τρόπος με τον οποίο έχει διαμορφωθεί μία δικτυακή συσκευή με σκοπό την προώθηση κίνησης, ονομάζεται πολιτική (policies). Οι πολιτικές αλλάζουν με την πάροδο του χρόνου, οι διαχειριστές αναδιαμορφώνουν τις δικτυακές συσκευές σαν απάντηση σε διάφορα συμβάντα όπως είναι για παράδειγμα, οι κυβερνοεπιθέσεις (cyber attacks),η βλάβη μιας συσκευής (device failure), η αλλαγή της κίνησης κ.α.

Στη συνέχεια παρουσιάζονται κάποια παραδείγματα διαμόρφωσης ενός δικτύου.

Δρομολόγηση: Η διαδικασία της δρομολόγησης αποτελεί μία από τις σημαντικότερες λειτουργίες ενός δικτύου. Σκοπός της είναι η μεταφορά πακέτων από ένα χρήστη σε κάποιον άλλο. Η κεφαλίδα ενός πακέτου περιλαμβάνει την διεύθυνση πηγής και προορισμού (IPv4,IPv6,Ethernet frame). Οι routers και τα switches κάνουν χρήση των διευθύνσεων προορισμού για την προώθηση των πακέτων στον τελικό αποδέκτη. Για παράδειγμα, σε ένα Ethernet δίκτυο τα πακέτα προωθούνται σύμφωνα με την φυσική διεύθυνση προορισμού (MAC). Ο παρακάτω κανόνας σε ένα Ethernet switch ορίζει ότι τα πακέτα με διεύθυνση προορισμού 01:00:00:00:00, προωθούνται στην θύρα 2 του switch.

match: dstMAC=01:00:00:00:00 action: fwd(2)

Σε ένα IP δίκτυο τα πακέτα προωθούνται σύμφωνα με την IP διεύθυνση προορισμού. Ο επόμενος κανόνας, ορίζει ότι πακέτα με IP διεύθυνση προορισμού από 10.0.0.1 έως 10.0.0.254, προωθούνται στην θύρα 5 του router.

match: dstIP=1.0.0.0/24 action: fwd(5)

Επίβλεψη: Η διαδικασία της επίβλεψης (Monitoring), συλλέγει στατιστικά στοιχεία από τις δικτυακές συσκευές. Έτσι, οι διαχειριστές κάνοντας χρήση των στατιστικών έχουν την δυνατότητα να αναγνωρίζουν συμβάντα που αφορούν την συμφόρηση ενός δικτύου, τον εντοπισμό σφαλμάτων σε αυτό, την βελτίωση της δρομολόγησης, τον εντοπισμό επιθέσεων κ.α. Για να γίνει αυτό, οι διαχειριστές χρειάζεται να διαμορφώσουν τους routers και τα switches. Για παράδειγμα, μία επιχείρηση η οποία θέλει να μετρήσει την web κίνηση στο δίκτυο της. Για να γίνει αυτό, ο διαχειριστής διαμορφώνει τους router και τα switches να αριθμούν πακέτα με αριθμό πρωτοκόλλου 6 (ο αριθμός πρωτοκόλλου 6 προσδιορίζει την χρήση του TCP πρωτοκόλλου) και αριθμό θύρας ίσο με 80 (ο αριθμός θύρας 80 χρησιμοποιείται από τους Web servers). Για την πραγματοποίηση της παραπάνω διαμόρφωσης χρειάζονται δύο κανόνες αντιστοίχισης, ένας για την θύρα προορισμού και ένας για την θύρα πηγής.

match: protocol=6, srcPort=80 action: count

match: protocol=6, dstPort=80 action: count

Firewall: Η ασφάλεια αποτελεί ίσως ένα από τα σημαντικότερα στοιχεία που πρέπει να διαθέτει μία επιχείρηση για την ομαλή της λειτουργία. Τα firewalls, εφαρμόζονται από τους διαχειριστές με σκοπό την προστασία του δικτύου τους. Στη συνέχεια διαμορφώνονται ώστε να ελέγχουν την εισερχόμενη και την εξερχόμενη κίνηση που επιτρέπεται να περάσει σε ένα δίκτυο. Για παράδειγμα, αν μία επιχείρηση θέλει να επιτρέπει την Web κίνηση εντός και εκτός του δικτύου, το firewall χρειάζεται τους δύο παρακάτω κανόνες:

match: protocol=6, srcPort=80 action: permit

match: protocol=6, dstPort=80 action: permit

Εκτός από χρονοβόρες οι παραπάνω διαδικασίες είναι και αρκετά πολύπλοκες. Επίσης, το δικτυακό περιβάλλον πρέπει να έχει την δυνατότητα να προσαρμόζεται εύκολα σε τυχόν αλλαγές της κίνησης (π.χ τις ώρες όπου παρατηρούνται μεγάλες απαιτήσεις για εύρος ζώνης και υπολογιστικούς πόρους) αλλά και να ανταποκρίνεται γρήγορα σε διάφορα σφάλματα που ενδέχεται να προκύψουν (π.χ η συμφόρηση σε ένα δίκτυο, η αποτυχία σύνδεσης). Όπως καταλαβαίνουμε, η εφαρμογή πολιτικών (policies) σε ένα τέτοιο δυναμικό περιβάλλον είναι ιδιαίτερα απαιτητική διαδικασία. Επιπλέον, στα σημερινά δίκτυα το επίπεδο του ελέγχου (control plane) το οποίο είναι υπεύθυνο για την λήψη αποφάσεων αλλά και για την διαχείριση της δικτυακής κίνησης, όπως και το επίπεδο των δεδομένων (data plane) είναι ενσωματωμένα σε κάθε συσκευή ξεχωριστά. Αυτός είναι ίσως και ο σημαντικότερος λόγος που στον τομέα της δικτύωσης ο ρυθμός της καινοτομίας κινείται με αργά βήματα.

Οι τωρινές αρχιτεκτονικές δικτύων δείχνουν να μην μπορούν ανταποκριθούν στις σύγχρονες ανάγκες των επιχειρήσεων, των παρόχων και των τελικών χρηστών. Έτσι, χάρη σε μια ευρεία προσπάθεια της βιομηχανίας με αιχμή του δόρατος τον Open Networking Foundation(ONF) **[1]** τα Δίκτυα Καθοριζόμενα από το Λογισμικό(SDN) έχουν σαν στόχο να αλλάξουν την παρούσα δικτυακή αρχιτεκτονική.

Το SDN **[2]** διαχωρίζει το επίπεδο του ελέγχου και το επίπεδο των δεδομένων των router και των switch, το αποτέλεσμα είναι οι δικτυακές συσκευές να μετατρέπονται σε απλές συσκευές προώθησης με το πεδίο του ελέγχου να μεταφέρεται σε ένα κεντρικό ελεγκτή, με αυτό τον τρόπο απλοποιείται, η επιβολή πολιτικών, ο έλεγχος των συσκευών καθώς και η επίβλεψη του δικτύου αλλά και της κίνησης. Η επικοινωνία του επιπέδου ελέγχου με το επίπεδο των δεδομένων πραγματοποιείται μέσω μίας διεπαφής (interface) μεταξύ του ελεγκτή και των συσκευών, το πιο αξιοσημείωτο παράδειγμα μίας τέτοιας διεπαφής είναι το OpenFlow.

Ο ONF (Open Networking Foundation) **[1]** είναι ένας οργανισμός μη κερδοσκοπικού χαρακτήρα που ηγείται της άφιξης του SDN στην αγορά καθώς και άλλων στοιχείων όπως είναι το OpenFlow πρωτόκολλο. Το OpenFlow είναι η πρώτη διεπαφή που σχεδιάστηκε αποκλειστικά για το SDN προσφέροντας υψηλή απόδοση, λεπτομερή παρακολούθηση της κίνησης μεταξύ διαφορετικών συσκευών διαφορετικών κατασκευαστών.

Το OpenFlow-based SDN χρησιμοποιείται σε διάφορες δικτυακές συσκευές αλλά και σε λογισμικό, προσφέροντας ουσιαστικά οφέλη στις επιχειρήσεις και στους παρόχους, κάποια από τα οποία είναι:

- Κεντρική διαχείριση και έλεγχος δικτυακών συσκευών διαφορετικών κατασκευαστών.
- Βελτιωμένη αυτοματοποίηση χρησιμοποιώντας APIs (Application Programming Interface) για τη αποσύνδεση των δικτυακών λεπτομερειών.
- Ταχεία καινοτομία μέσω της δυνατότητας μεταφοράς νέων δικτυακών ικανοτήτων και υπηρεσιών χωρίς την ανάγκη της διαμόρφωσης των συσκευών ή την αναμονή για καινούρια έκδοση (update) από τον κατασκευαστή.
- Προγραμματισιμότητα (programmability) για τους διαχειριστές των δικτύων, για τις επιχειρήσεις, για ανεξάρτητους κατασκευαστές λογισμικού καθώς και τους χρήστες χρησιμοποιώντας κοινά προγραμματιστικά περιβάλλοντα κάτι που δίνει την ευκαιρία για αύξηση κερδών.
- Αυξημένη δικτυακή αξιοπιστία και ασφάλεια σαν αποτέλεσμα της κεντρικής διαχείρισης και αυτοματοποίησης των δικτυακών συσκευών, την επιβολή ομοιόμορφων πολιτικών και λιγότερων λαθών κατά την διαμόρφωση των συσκευών.
- Καλύτερη παροχή υπηρεσιών στους τελικούς χρήστες από την στιγμή που οι εφαρμογές αξιοποιούν τις πληροφορίες για την κατάσταση του κεντρικού δικτύου ώστε να προσαρμόζουν την συμπεριφορά του δικτύου σύμφωνα με τις ανάγκες των τελικών χρηστών.

1.2 Η Ανάγκη Για Μια Καινούρια Αρχιτεκτονική

Η αύξηση των κινητών συσκευών σε συνδυασμό με τις εφαρμογές που τις συνοδεύουν, η εικονοικοποίηση(Virtualization) των Servers και η άφιξη των cloud υπηρεσιών είναι οι λόγοι που οδηγούν την δικτυακή βιομηχανία να επανεξετάσει την παραδοσιακή δικτυακή αρχιτεκτονική. Αυτή η σχεδίαση είχε νόημα όταν η υπολογιστική client-server επικρατούσε, αλλά μία τέτοια στατική αρχιτεκτονική είναι απρόσφορη στην δυναμική υπολογιστική και στις ανάγκες αποθήκευσης των σημερινών data centers. Κάποιες από τις τάσεις της υπολογιστικής που απαιτούν την αναδιαμόρφωση της τρέχουσας αρχιτεκτονικής είναι:

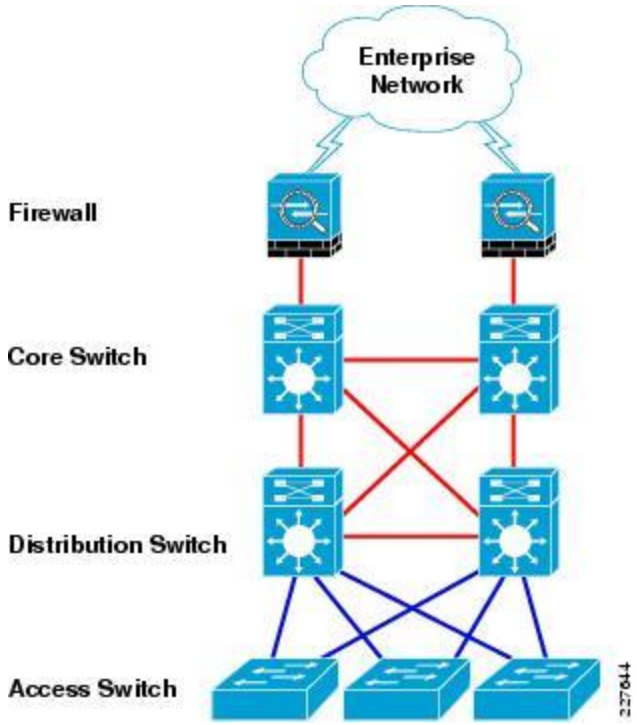
- **Αλλαγή στον τρόπο κίνησης:** Στα Data Centers των εταιριών ο τρόπος κίνησης των δεδομένων έχει αλλάξει σημαντικά. Σε αντίθεση με τις client-server εφαρμογές όπου ο κύριος όγκος της επικοινωνίας πραγματοποιείται μεταξύ ενός client και ενός server, οι τωρινές εφαρμογές έχουν πρόσβαση σε διαφορετικές βάσεις δεδομένων και servers δημιουργώντας μια κίνηση της μορφής east-west πριν επιστρέψουν δεδομένα στον τελικό χρήστη σε αντίθεση με την κλασική north-south κίνηση. Ταυτόχρονα η πρόσβαση των χρηστών σε εταιρικές εφαρμογές μέσω οποιαδήποτε τύπου συσκευής συνδεόμενοι σε ακαθόριστο χρόνο και σημείο αλλάζει επίσης την μορφή της δικτυακής κίνησης. Επιπρόσθετα οι διαχειριστές των εταιρικών Data Center χρησιμοποιούν ένα υπολογιστικό μοντέλο που μπορεί να περιλαμβάνει ένα ιδιωτικό cloud ή ένα δημόσιο ή ακόμη και τα δύο, προσθέτοντας κίνηση κατά μήκους του δικτύου.

***East-west traffic**: Με τον συγκεκριμένο όρο εννοούμε την κίνηση στο ίδιο κέντρο δεδομένων δηλαδή στην κίνηση μεταξύ ενός διακομιστή με ένα άλλο διακομιστή, ενώ με τον όρο **North-South traffic** εννοούμε την κίνηση μεταξύ client-server, δηλαδή την κίνηση μεταξύ του κέντρου δεδομένων με το υπόλοιπο δίκτυο (οτιδήποτε έξω από το κέντρο δεδομένων).*

- **Η εμπορευματοποίηση του IT :** Οι χρήστες που χρησιμοποιούν κινητές συσκευές όπως smartphones, tablets, notebooks με σκοπό την πρόσβαση σε ένα εταιρικό δίκτυο, θέτοντας το τμήμα IT υπό πίεση διότι πρέπει να ικανοποιήσει τις προσωπικές συσκευές του κάθε χρήστη και ταυτόχρονα να παρέχει ασφάλεια.
- **Η Άνοδος των cloud υπηρεσιών :** Οι επιχειρήσεις έχουν απροσδόκητα αγκαλιάσει τις cloud υπηρεσίες. Απαιτούν ευκολία πρόσβασης σε εφαρμογές, σε υποδομή και άλλους πόρους του IT. Επιπρόσθετα η ανάθεση των cloud υπηρεσιών οφείλει να υλοποιείται σε ένα περιβάλλον αυξημένης ασφάλειας και ελέγχου.
- **Big Data σημαίνει περισσότερο bandwidth:** Ο χειρισμός των σημερινών "big data" απαιτεί μαζική παράλληλη επεξεργασία σε χιλιάδες server οι οποίοι πρέπει να είναι άμεσα συνδεδεμένοι μεταξύ τους. Η άνοδος των big datasets δημιουργεί συνεχώς απαιτήσεις για επιπλέον χωρητικότητα στα data centers.

Επίσης, αρκετά συμβατικά δίκτυα είναι ιεραρχικά, χτισμένα σε βαθμίδες των Ethernet switches τοποθετημένα σε δομή δένδρου, όπως φαίνεται και από την εικόνα 2. Πιο συγκεκριμένα, τα switch πρόσβασης (access switch) συνδέονται απευθείας με τις τερματικές συσκευές (είτε πρόκειται για υπολογιστές, είτε για servers) παρέχοντας τους πρόσβαση στο LAN. Τα switch πρόσβασης προωθούν την κίνηση από και προς τις τερματικές συσκευές στα οποία είναι συνδεδεμένα και δημιουργούν μία τοπολογία αστέρα. Τα switch διανομής (distribution switch) παρέχουν μία διαδρομή την οποία χρησιμοποιούν τα switch πρόσβασης για να προωθήσουν την κίνηση μεταξύ τους. Τα switch πρόσβασης συνήθως συνδέονται σε παραπάνω από ένα switch διανομής, αυτό συμβαίνει συνήθως σε περιπτώσεις όπου όταν μία σύνδεση μεταξύ ενός switch πρόσβασης και ενός switch διανομής αποτύχει για διάφορους λόγους, να υπάρχει τουλάχιστον μία εναλλακτική διαδρομή για την προώθηση της κίνησης. Τα switch κορμού (core switches) είναι υπεύθυνα για την γρήγορη και αξιόπιστη μεταφορά δεδομένων σε ένα δίκτυο. Ο λόγοι που συνήθως τα χρησιμοποιούμε είναι για να επιτύχουμε υψηλό ποσοστό μεταφοράς δεδομένων, για να έχουμε χαμηλά επίπεδα καθυστέρησης και για να εξασφαλίσουμε αξιοπιστία μέσω πολλαπλών διαδρομών. Όταν για παράδειγμα μία σύνδεση αντιμετωπίζει κάποιο πρόβλημα, τότε η συσκευή μπορεί να ανακαλύψει γρήγορα μία νέα διαδρομή. Τα firewalls όπως αναφέραμε και στο **1.1,** είναι υπεύθυνα για το είδος της κίνησης που επιτρέπεται να εισέλθει και να εξέλθει σε ένα δίκτυο.

Big Data*: Ο όρος Big Data συχνά αναφέρεται και σαν 3vs, εννοώντας το μεγάλο όγκο των δεδομένων (volume of data), την ποικιλία διαφορετικών τύπων δεδομένων (variety of data types) και την ταχύτητα με την οποία πρέπει να επεξεργαστούν τα δεδομένα (velocity of processing data). Τα Big Data καταφθάνουν από διάφορες πηγές, όπως είναι για παράδειγμα οι αισθητήρες(sensors), οι κινητές συσκευές κ.α.Για να εξάγουμε την αξία που έχουν τα Big Data, χρειάζεται βέλτιστη επεξεργαστική ισχύ, δυνατότητες ανάλυσης και άλλες δεξιότητες.*



Εικόνα 2:Τοπολογία ενός δικτύου σε δομή δένδρου (Three-tier campus topology).

1.3 Περιορισμοί Των Τρέχουσων Δικτυακών Τεχνολογιών

Τα IT τμήματα των εταιριών προσπαθούν να αποσπάσουν τα μέγιστα από το δίκτυο τους χρησιμοποιώντας χειροκίνητες διαδικασίες καθώς και εργαλεία διαχείρισης σε επίπεδο συσκευής. Οι πάροχοι αντιμετωπίζουν της ίδιες προκλήσεις καθώς οι απαιτήσεις για κινητικότητα και περισσότερο εύρος ζώνης από τους τελικούς χρήστες συνεχώς αυξάνει. Το αποτέλεσμα είναι πως η υπάρχουσα αρχιτεκτονική δεν επαρκεί καθώς δεν σχεδιάστηκε με σκοπό να ικανοποιήσει τις σημερινές ανάγκες. Μερικοί περιορισμοί που συναντάμε στα τωρινά δίκτυα είναι:

- **Η πολυπλοκότητα που οδηγεί σε στάση**

  Η δικτυακή τεχνολογία μέχρι και σήμερα βασίζεται σε μία σειρά διακριτών πρωτοκόλλων σχεδιασμένων να συνδέουν τους χρήστες, αξιόπιστα, σε τυχαίες αποστάσεις, ταχύτητες συνδέσεων και τοπολογίες. Για να ικανοποιήσει αυτές τις ανάγκες τα τελευταία χρόνια η βιομηχανία έχει εξελίξει διάφορα δικτυακά πρωτόκολλα με στόχο την υψηλότερη απόδοση, τη βελτίωση της αξιοπιστίας και της ασφάλειας καθώς και την παροχή ευρύτερης συνδεσιμότητας .

  Παρόλα αυτά, τα συγκεκριμένα πρωτόκολλα έχουν σχεδιαστεί ώστε να επιλύουν ένα συγκεκριμένο πρόβλημα. Αυτό έχει σαν αποτέλεσμα να οδηγούμαστε στην σημερινή πολυπλοκότητα των δικτύων. Για παράδειγμα για να προσθέσουμε ή να μετακινήσουμε μία συσκευή το IT τμήμα πρέπει να διαμορφώσει όλα τα switches, τους routers, τα firewalls, τα ACLs (Access Control Lists),τα VLANs (Virtual Local Area Networks), αλλά και το Qos (Quality of Service).

Η στατική φύση των δικτύων έρχεται σε αντίθεση με την δυναμική φύση του περιβάλλοντος των server, όπου η εικονικοποίηση (*virtualization*) του server έχει αυξήσει σημαντικά των αριθμό των χρηστών οι οποίοι μπορούν να εξυπηρετηθούν ταυτόχρονα αλλάζοντας επίσης τις υποθέσεις που μπορούμε να κάνουμε αναφορικά με την φυσική τοποθεσία των χρηστών. Πριν την εικονικοποίηση, οι εφαρμογές βρισκόντουσαν σε έναν μόνο server και η ανταλλαγή κίνησης γινόταν με επιλεγμένους client. Οι σημερινές εφαρμογές διαμοιράζονται από τους Servers δια μέσου πολλαπλών εικονικών συσκευών (VMs) οι οποίες ανταλλάσσουν ροές κίνησης μεταξύ τους. Η χρήση εικονικών συσκευών βελτιστοποιεί και εξισορροπεί τον όγκο εργασίας για τους server.

Εκτός από την υιοθέτηση τεχνολογιών εικονικοποίησης, πολλές εταιρίες σήμερα, λειτουργούν ένα δίκτυο για φωνή, δεδομένα και video. Παρόλο που τα υπάρχοντα δίκτυα μπορούν να παρέχουν διαφοροποιημένα επίπεδα QoS, για διαφορετικές εφαρμογές, η παροχή των πόρων και η διαμόρφωση των συσκευών, γίνεται συνήθως χειροκίνητα. Το IT πρέπει να διαμορφώσει κάθε εξοπλισμό ξεχωριστά, ανάλογα με τον κατασκευαστή. Εξαιτίας της στατικής φύσης του, το δίκτυο δεν μπορεί να προσαρμόζεται δυναμικά στις αλλαγές της κίνησης, των εφαρμογών και των απαιτήσεων των χρηστών.

- **Ασυνεπείς Πολιτικές**
  Για την εφαρμογή μιας ευρείας δικτυακής πολιτικής το IT τμήμα πρέπει να διαμορφώσει χιλιάδες συσκευές και μηχανισμούς. Για παράδειγμα, κάθε φορά που μία καινούρια εικονική μηχανή προστίθεται στο δίκτυο μπορεί να πάρει ώρες έως και μέρες για το IT να αναδιαμορφώσει το δίκτυο, για παράδειγμα τα ACLs κατά μήκους του δικτύου. Αυτή η πολυπλοκότητα δεν επιτρέπει στο IT να εφαρμόσει συνεπής πολιτικές πρόσβασης, ασφάλειας και QoS. Έτσι οι επιχειρήσεις είναι ευάλωτες σε κινδύνους ασφάλειας.
- **Αδυναμία Κλιμάκωσης**
  Οι απαιτήσεις στα κέντρα δεδομένων (*Data Centers*) έχουν μεγαλώσει ραγδαία το ίδιο ισχύει βέβαια και στην πλευρά του δικτύου. Παρόλα αυτά το δίκτυο γίνεται ολοένα και πιο σύνθετο με την προσθήκη εκατοντάδων δικτυακών συσκευών, που πρέπει να διαμορφωθούν και να τεθούν υπό διαχείριση.
- **Εξάρτηση από τον κατασκευαστή**
  Οι πάροχοι και οι επιχειρήσεις αναζητούν να εφαρμόσουν νέες τεχνικές και υπηρεσίες σαν απάντηση στις ανάγκες των επιχειρήσεων και των χρηστών για περισσότερους δικτυακούς πόρους. Όλο αυτό το εγχείρημα παρεμποδίζεται από τον κύκλο ζωής του εξοπλισμού του εκάστοτε κατασκευαστή ο οποίος κυμαίνεται στα τρία χρόνια ή και περισσότερο. Η έλλειψη προτύπων και ανοιχτών διεπαφών περιορίζει την ικανότητα των παρόχων να προσαρμόσουν το δίκτυο στις ατομικές τους ανάγκες.

Στη συνέχεια παρατίθεται ένα παράδειγμα ενός δικτύου στο οποίο χρειάζεται να χρησιμοποιηθεί μία νέα εφαρμογή για τις ανάγκες μίας επιχείρησης. Αναφέρουμε ποιες δικτυακές συσκευές πρέπει να αναδιαμορφωθούν για να λειτουργήσει σωστά η εφαρμογή ώστε οι τελικοί χρήστες να μπορούν να έχουν πρόσβαση σε αυτή. Παράλληλα, σκοπός του παραδείγματος είναι να δείξει τους περιορισμούς που συναντάμε στα σημερινά δίκτυα όπως έχει αναφερθεί ήδη και πιο πάνω.

Στην εικόνα 2 του **1.2**, φαίνεται η τοπολογία ενός εταιρικού δικτύου σε δομή δένδρου και οι διαφορετικές λειτουργίες που επιτελούν τα switch. Ας πάρουμε για παράδειγμα το εξής σενάριο: ένα από τα switch πρόσβασης της παραπάνω εικόνας συνδέεται σε έναν server, για παράδειγμα έναν VMware server στον οποίο τρέχουν κάποιες εικονικές μηχανές (Virtual Machines). Το switch πρόσβασης συνδέεται σε τουλάχιστον ένα switch διανομής και αυτό με την σειρά του είναι συνδεδεμένο σε ένα switch κορμού. Επίσης, γίνεται χρήση των firewalls τα οποία παρέχουν την ενδεδειγμένη ασφάλεια στο δίκτυο. Ακόμη, αν και δεν απεικονίζεται στην εικόνα 2 η εταιρεία πρέπει να έχει εγκατεστημένους routers για την δρομολόγηση της κίνησης από και προς το εταιρικό δίκτυο,στην πλειοψηφία των περιπτώσεων τουλάχιστον δύο (για περιπτώσεις που όταν ο ένας router αντιμετωπίζει κάποιο πρόβλημα να υπάρχει τουλάχιστον μία εναλλακτική λύση για την δρομολόγηση των πακέτων). Ας υποθέσουμε ότι η εταιρεία χρειάζεται να χρησιμοποιήσει μία νέα εφαρμογή η οποία απαιτεί την εγκατάσταση τεσσάρων νέων εικονικών μηχανών στον VMware server. Για λόγους ασφαλείας, η κάθε εικονική μηχανή πρέπει να είναι σε διαφορετικό VLAN. Μερικά από τα πράγματα τα οποία πρέπει να διαμορφωθούν στο δίκτυο για να λειτουργήσει η νέα εφαρμογή είναι τα εξής:

- Τα καινούρια VLANs πρέπει να δημιουργηθούν σε όλα τα switches.
- Πρέπει να δημιουργηθούν τέσσερα νέα υποδίκτυα για κάθε VLAN.
- Τα firewalls πρέπει να διαμορφωθούν ώστε να επιτρέπουν την πρόσβαση στα καινούρια υποδίκτυα αλλά και στην καινούρια εφαρμογή.
- Οι routers κάνοντας χρήση ενός πρωτοκόλλου εσωτερικής δρομολόγησης π.χ RIPv2 (Routing Information Protocol), OSPF (Open Shortest Path First), EIGRP (Enhanced Interior Gateway Routing Protocol) πρέπει να ανταλλάξουν πληροφορίες μεταξύ τους για την ύπαρξη των τεσσάρων νέων υποδικτύων.
- Οι διεπαφές που συνδέουν τα switch μεταξύ τους πρέπει να διαμορφωθούν ώστε να επιτρέπουν να περνάει η κίνηση από τα τέσσερα καινούρια VLANs που θα προστεθούν.
- Τα switch θα πρέπει να κάνουν χρήση του STP (Spanning Tree Protocol) πρωτοκόλλου για την βελτίωση της αποδοτικότητας του εσωτερικού δικτύου αλλά και για την αποφυγή εσωτερικών βρόχων.
- Οι routers θα χρειαστεί να αναδιαμορφώσουν τις λίστες ελέγχου πρόσβασης που έχουν ή να δημιουργηθούν καινούριες.

Όπως έχει αναφερθεί ήδη, για την σωστή λειτουργία του παραπάνω παραδείγματος είναι απαραίτητο να διαμορφωθούν όλες οι δικτυακές συσκευές ξεχωριστά. Η συνηθέστερη μέθοδος είναι κάνοντας χρήση του CLI (Command Line Interface) της εκάστοτε συσκευής, η οποία εκτός του ότι είναι μία αργή διαδικασία εμπεριέχει και τον κίνδυνο πιθανών λαθών, διότι γίνεται χειροκίνητα. Έτσι, το τελικό συμπέρασμα του παραπάνω παραδείγματος είναι ότι ενώ

χρειάζεται μερικά λεπτά για την ενεργοποίηση των νέων εικονικών μηχανών, ενδέχεται να χρειαστούν αρκετές ώρες από τμήμα του IT για να προετοιμάσει το δίκτυο.

Αυτή η αναντιστοιχία μεταξύ των απαιτήσεων της αγοράς και των ικανοτήτων του δικτύου έχει οδηγήσει την βιομηχανία στην αναζήτηση νέων λύσεων και στην δημιουργία της SDN αρχιτεκτονικής.

1.4 Εισαγωγή Στην SDN δικτύωση

Όπως αναφέρθηκε και στο **1.1** ένα παραδοσιακό δίκτυο αποτελείται από συσκευές οι οποίες πραγματοποιούν μία συγκεκριμένη λειτουργία σε αυτό, οποιαδήποτε λειτουργία πραγματοποιείται από τις συσκευές, κατατάσσεται σε ένα συγκεκριμένο επίπεδο. Πιο συγκεκριμένα, κάθε συσκευή έχει τρία επίπεδα: το επίπεδο ελέγχου, το επίπεδο των δεδομένων και το επίπεδο της διαχείρισης (management plane).

Το επίπεδο των δεδομένων αναφέρεται στις εργασίες που πραγματοποιεί μία συσκευή για την προώθηση ενός μηνύματος. Με άλλα λόγια οτιδήποτε σχετίζεται με την λήψη, την επεξεργασία και την προώθηση των δεδομένων (είτε τα δεδομένα αυτά είναι ένα frame, είτε ένα πακέτο είτε ένα μήνυμα) είναι μέρος του επιπέδου των δεδομένων της συσκευής. Ας πάρουμε σαν παράδειγμα την διαδικασία της δρομολόγησης ενός πακέτου μεταξύ ενός host και ενός server, η IP διεύθυνση του οποίου βρίσκεται σε ένα υποδίκτυο διαφορετικό από τον host. Αρχικά ο host δημιουργεί το πακέτο που θέλει να στείλει στον server, συγκρίνει την δική του IP με την IP διεύθυνση του server και διαπιστώνει ότι η IP διεύθυνση προορισμού του πακέτου ανήκει σε δίκτυο διαφορετικό από αυτόν. Στη συνέχεια, παίρνει την απόφαση να δρομολογήσει το πακέτο στον προεπιλεγμένο του router (default gateway), ο router με την σειρά του επεξεργάζεται το λαμβανόμενο πακέτο (συγκρίνει την IP διεύθυνση προορισμού με καταχωρήσεις που έχει στον πίνακα δρομολόγησης), παίρνει μία απόφαση δρομολόγησης και προωθεί το πακέτο στον επόμενο router (next-hop router) μέχρις ότου αυτό να καταλήξει στον τελικό προορισμό.

Η ακόλουθη λίστα παρουσιάζει μερικές από τις λειτουργίες για τις οποίες είναι υπεύθυνο το επίπεδο των δεδομένων:

- Ενθυλάκωση, από-θυλάκωση (encapsulate,de-encapsulate) πακέτων.
- Προσθήκη ή αφαίρεση κεφαλίδων, για παράδειγμα η 802.1Q κεφαλίδα (χρησιμοποιείται σε περιπτώσεις που έχουμε πάνω από ένα VLAN (Virtual Local Area Network) διαμορφωμένα και θέλουμε τα switch να γνωρίζουν σε ποιο VLAN απευθύνεται το πακέτο, επίσης επιτρέπει στους routers να δρομολογούν την κίνηση μεταξύ διαφορετικών VLAN.)
- Αντιστοίχιση MAC διευθύνσεων για προώθηση.
- Αντιστοίχιση IP διευθύνσεων στον πίνακα δρομολόγησης των router.
- Αλλαγή της διεύθυνσης πηγής ή της διεύθυνσης προορισμού όταν γίνεται χρήση του πρωτοκόλλου NAT (Network Address Translation).
- Απόρριψη της κίνησης εξαιτίας της παρουσίας λίστας ελέγχου πρόσβασης (Access Control List).

Οι παραπάνω εργασίες πρέπει να πραγματοποιούνται από το επίπεδο των δεδομένων όσο το δυνατόν γρηγορότερα. Για το λόγο αυτό, γίνεται η χρήση εξειδικευμένου υλικού όπως είναι οι πίνακες ASIC (Application Specific Intergrated Circuit) ή TCAM (Temporary Content Addressable Memory) πίνακες.

Επίσης, το επίπεδο των δεδομένων χρειάζεται να γνωρίζει ένα συγκεκριμένο είδος πληροφορίας από πριν ώστε να λειτουργεί σωστά. Για παράδειγμα, οι routers απαιτείται να έχουν πληροφορίες δρομολόγησης (IP routes) στον πίνακα τους (routing table) πριν το επίπεδο των δεδομένων αναλάβει την μετάδοση ενός πακέτου. Τα switches με την σειρά τους πρέπει να διαθέτουν καταχωρήσεις MAC διευθύνσεων στον πίνακα τους (MAC address table) πριν λάβουν την απόφαση για την προώθηση ενός frame.

Κατά μία άποψη, οι πληροφορίες που παρέχονται στο επίπεδο των δεδομένων ελέγχουν ταυτόχρονα και την λειτουργία του. Ο router δεν έχει την δυνατότητα να προωθήσει πακέτα εάν δεν έχει διαδρομές στον πίνακα του. Όταν το επίπεδο των δεδομένων προσπαθήσει να βρει μία καταχώρηση που να ταιριάζει με την διεύθυνση του πακέτου που έχει λάβει και δεν βρει καμία τότε απορρίπτει το πακέτο. Αντίθετα, αν ο router έχει διαδρομές στον πίνακα του, το επίπεδο των δεδομένων μπορεί να προωθήσει πακέτα, έτσι οι παράγοντες που είναι υπεύθυνοι για το περιεχόμενο ενός πίνακα δρομολόγησης είναι οι διάφορες διαδικασίες του επιπέδου ελέγχου.

Συγκεκριμένα , με τον όρο επίπεδο ελέγχου ορίζεται οποιαδήποτε δράση ελέγχει το επίπεδο των δεδομένων. Οι περισσότερες από αυτές τις ενέργειες έχουν να κάνουν με την δημιουργία των πινάκων που χρησιμοποιούνται από το επίπεδο των δεδομένων όπως είναι, οι πίνακες δρομολόγησης, οι ARP πίνακες και οι MAC πίνακες που χρησιμοποιούνται από τα switch.

Ενδεικτικά μερικές από τις εργασίες για τις οποίες είναι υπεύθυνο το επίπεδο ελέγχου είναι:

- Η εκμάθηση των MAC διευθύνσεων ώστε το switch να μπορεί να έχει ένα ακριβή πίνακα MAC διευθύνσεων.
- Εκτέλεση του STP πρωτοκόλλου για την δημιουργία τοπολογίας χωρίς βρόχους.
- Δημιουργία ARP (Address Resolution Protocol) πινάκων.
- Εκτέλεση πρωτοκόλλων δρομολόγησης όπως το OSPF, EIGRP ,RIPv2 κ.α.

Όπως έχει ήδη αναφέρει, οι λειτουργίες που πραγματοποιεί το επίπεδο ελέγχου έχουν άμεσο αντίκτυπο στην συμπεριφορά του επιπέδου των δεδομένων. Το τρίτο και τελευταίο επίπεδο των συσκευών, αυτό της διαχείρισης πραγματοποιεί λειτουργίες οι οποίες όμως δεν έχουν άμεση επίδραση σε αυτό των δεδομένων. Το επίπεδο της διαχείρισης επιτρέπει την απομακρυσμένη πρόσβαση και διαχείριση των δικτυακών συσκευών. Για παράδειγμα με την χρήση των πρωτοκόλλων όπως είναι το Telnet ή το SSH (τα οποία είναι τα πιο δημοφιλή πρωτόκολλα απομακρυσμένης διαχείρισης) μπορούμε να έχουμε απομακρυσμένη πρόσβαση σε κάθε συσκευή για την διαχείριση, τον έλεγχο και την διαμόρφωσης της.

Στη ουσία τα τρία επίπεδα που αναφέρθηκαν παραπάνω λειτουργούν σε κάθε συσκευή ξεχωριστά. Επιπλέον, η κάθε δικτυακή συσκευή περιορίζεται στο να έχει πληροφορίες μόνο για τους γειτονικούς κόμβους με τους οποίους επικοινωνεί, με αλλά λόγια δεν υπάρχει μία κεντρική συσκευή που να έχει μία συνολική εικόνα για την κατάσταση του δικτύου ή να έχει τον πλήρη

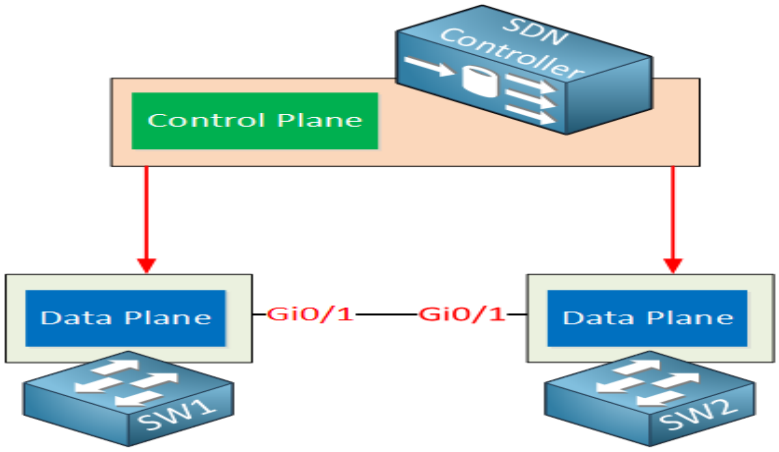
έλεγχο αυτού. Εξαίρεση αποτελούν (για όσους είναι εξοικειωμένοι με την ασύρματη δικτύωση) οι ασύρματοι ελεγκτές **[8]**.

Κατά την διαμόρφωση ενός ασύρματου δικτύου όλες οι ρυθμίσεις πραγματοποιούνται στον ασύρματο ελεγκτή, εκεί μπορούμε να ορίσουμε το όνομα του ασύρματου δικτύου (SSID), να δηλώσουμε ποιοι χρήστες επιτρέπεται να συνδεθούν και ποιοι θα απορρίπτονται αλλά και το ποιοι θα είναι οι κωδικοί πρόσβασης. Ο ελεγκτής ελέγχει και διαμορφώνει τα σημεία πρόσβασης (access points), έτσι δεν χρειάζεται να διαμορφωθεί κάθε σημείο πρόσβασης ξεχωριστά, αυτή η λειτουργία γίνεται μέσω του ασύρματου ελεγκτή.

Η αρχιτεκτονική των SDN **[2]** δικτύων περιγράφεται από μία δομή τριών επιπέδων: το επίπεδο της υποδομής, το επίπεδο του ελέγχου και το επίπεδο των εφαρμογών. Στη συνέχεια παρουσιάζεται το κάθε επίπεδο ξεχωριστά:

- **Επίπεδο της Υποδομής (Infrastructure layer)**
  Είναι το επίπεδο όπου βρίσκεται το υλικό (hardware) και υλοποιείται η φυσική διασύνδεσή του. Στις συσκευές υλικού εκτελείται λογισμικό, το οποίο παρέχει μία διεπαφή ελέγχου του επιπέδου των δεδομένων (Southbound API). Αυτή η διεπαφή χρησιμοποιείται για επικοινωνία με το επίπεδο ελέγχου.
- **Επίπεδο Ελέγχου (Control layer)**
  Αποτελεί ίσως το σημαντικότερο κομμάτι της αρχιτεκτονικής. Στο επίπεδο ελέγχου υπάρχει ένας ελεγκτής, ο οποίος επικοινωνεί με όλες τις δικτυακές συσκευές της υποδομής και ταυτόχρονα παρακολουθεί την τοπολογία. Ο ελεγκτής πραγματοποιεί ανταλλαγή πληροφοριών, σχετικά με την κατάσταση του δικτύου, με το ανώτερο επίπεδο αυτό της εφαρμογής μέσω ενός Northbound API. Ο ελεγκτής μεταφράζει τις εντολές προς τις συσκευές δικτύωσης, έτσι ώστε αυτές να έχουν την επιθυμητή συμπεριφορά και απόκριση.
- **Επίπεδο Εφαρμογών (Application layer)**
  Είναι το επίπεδο, όπου ορίζονται όλα τα χαρακτηριστικά , οι υπηρεσίες και οι πολιτικές. Οι εφαρμογές ζητούν πληροφορίες, σχετικά με τις συσκευές δικτύωσης και την τοπολογία του δικτύου, ώστε να δρουν αναλόγως. Επιπλέον, μπορούν να λαμβάνουν αποφάσεις με βάση τις αλλαγές στο δίκτυο. Κάθε φορά που αλλάζει η τοπολογία του δικτύου ή κάποια πολιτική, οι πολιτικές μπορούν να αλλάξουν δυναμικά την συμπεριφορά του δικτύου, από ένα μοναδικό σημείο.

Ο ελεγκτής αναλαμβάνει την εκτέλεση των λειτουργιών που αφορούν το επίπεδο ελέγχου της κάθε δικτυακής συσκευής, όπως είναι για παράδειγμα, η δρομολόγηση, η επιβολή πολιτικών, η εφαρμογή ελέγχου ασφάλειας κ.α. Ο ελεγκτής μπορεί να είναι μία φυσική συσκευή υλικού ή μία εικονική μηχανή εγκατεστημένη σε κάποιο server.



Εικόνα 3: Διαχωρισμός του Control και του Data plane και επικοινωνία του ελεγκτή με το switch

Στην παραπάνω εικόνα φαίνεται ο ελεγκτής ο οποίος όπως είπαμε είναι υπεύθυνος για το επίπεδο ελέγχου της κάθε συσκευής. Τα switch μετατρέπονται πλέον σε συσκευές προώθησης οι οποίες διαθέτουν μόνο επίπεδο δεδομένων, ο ελεγκτής είναι υπεύθυνος για την ενημέρωση του πεδίου των δεδομένων των switch με πληροφορίες από το επίπεδο ελέγχου του. Ένα από τα πλεονεκτήματα της συγκεκριμένης αρχιτεκτονικής είναι ότι δεν χρειάζεται να διαμορφώσουμε κάθε συσκευή ξεχωριστά, αυτό επιτυγχάνεται με την χρήση μίας μόνο συσκευής.

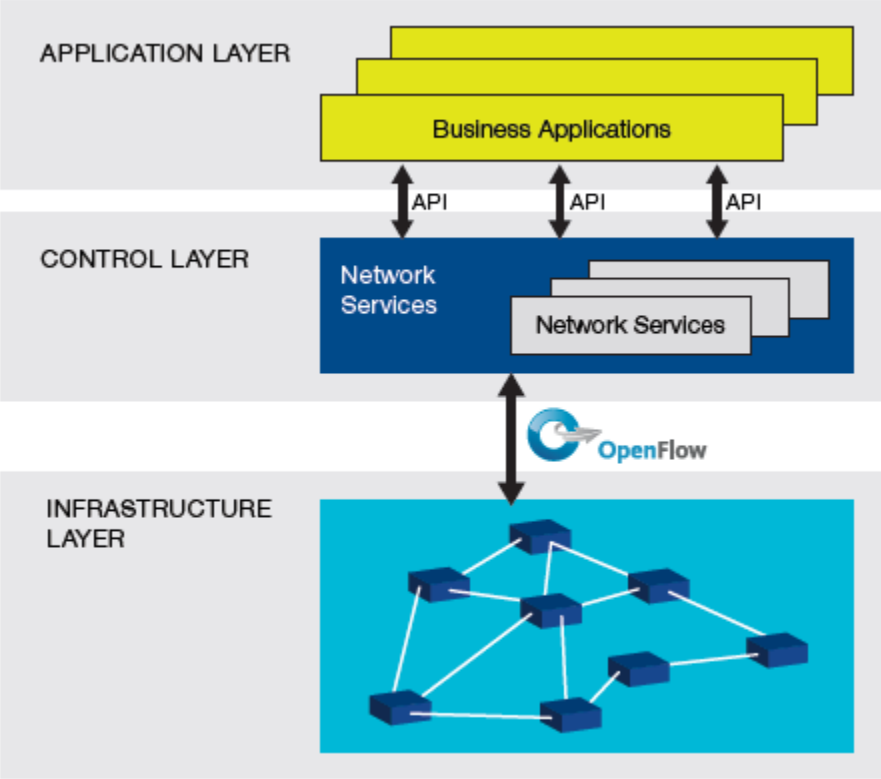
Μεταξύ των τριών επιπέδων υπάρχουν οι Διεπαφές Προγραμματισμού Εφαρμογών (APIs), οι οποίες παρέχουν τα απαραίτητα εργαλεία επικοινωνίας.

Ο ελεγκτής χρησιμοποιεί δύο διεπαφές (κυρίως), την νότια διεπαφή (southbound interface) και την βόρεια διεπαφή (northbound interface). Η νότια διεπαφή χρησιμοποιείται για τον προγραμματισμό του επιπέδου των δεδομένων των δικτυακών συσκευών. Η συγκεκριμένη διεπαφή δεν είναι μία φυσική διεπαφή αλλά μία διεπαφή λογισμικού συνήθως ένα API, και όπως αναφέρθηκε και πιο πάνω και όπως αναλύεται εκτενώς στο κεφάλαιο 2 η δημοφιλέστερη είναι το OpenFlow.

Η βόρεια διεπαφή επιτρέπει στον διαχειριστή του δικτύου να έχει πρόσβαση στον ελεγκτή, να τον διαμορφώσει ή και να ανακτήσει πληροφορία από αυτόν. Αυτό μπορεί να γίνει με διάφορα APIs τα οποία επιτρέπουν στις εφαρμογές να έχουν πρόσβαση στον ελεγκτή. Με την βόρεια διεπαφή καθίσταται δυνατή η αυτοματοποίηση της διαχείρισης του δικτύου αλλά και η συγγραφή προγραμμάτων (scripts). Μερικά παραδείγματα μπορεί να είναι:

- Η Καταχώρηση πληροφοριών σχετικά με τις συσκευές του δικτύου.
- Η Εμφάνιση της κατάστασης όλων των φυσικών διεπαφών στο δίκτυο.
- Η Προσθήκη ενός νέου VLAN σε όλα τα switch.
- Η Εμφάνιση της τοπολογίας ολόκληρου του δικτύου.
- Η Αυτόματη διαμόρφωση IP διευθύνσεων, δρομολόγησης, λιστών ελέγχου πρόσβασης όταν δημιουργείται μία νέα εικονική μηχανή.

Επίσης, στην κατανεμημένη SDN αρχιτεκτονική γίνεται η χρήση της ανατολικής (Eastbound) και της δυτικής (Westbound) διεπαφής. Στις λειτουργίες του περιλαμβάνεται η ανταλλαγή δεδομένων μεταξύ των ελεγκτών, ορισμένες δυνατότητες παρακολούθησης και ενημέρωσης όπως και η υλοποίηση αλγόριθμων συνοχής δεδομένων.



Εικόνα 4: Ενδεικτικό σχήμα της SDN αρχιτεκτονικής

Ένα από τα σημαντικότερα πλεονεκτήματα είναι ότι οι διαχειριστές ενός δικτύου μπορούν να διαμορφώσουν τις συσκευές μέσω προγραμμάτων γραμμένων σε γλώσσα υψηλού επιπέδου (Java,Python), αντί να γράφουν δεκάδες εντολές για να διαμορφώσουν κάθε συσκευή ξεχωριστά. Επίσης, με την μεταφορά της δικτυακής ευφυίας σε ένα κεντρικό σημείο είναι εφικτή η δημιουργία νέων εφαρμογών και δικτυακών υπηρεσιών σε λίγες μόνο ώρες ή και ημέρες, αντί για εβδομάδες ή και μήνες που απαιτείται τώρα. Με τον τρόπο αυτό, δίνεται η δυνατότητα στις επιχειρήσεις να δημιουργήσουν προγράμματα που προσαρμόζονται στις ατομικές τους ανάγκες για την διαχείριση του δικτύου τους, αντί να περιμένουν από τον κατασκευαστή της εκάστοτε συσκευής να διανέμει μία αναβάθμιση λογισμικού.

Επιπρόσθετα, η SDN αρχιτεκτονική υποστηρίζει ένα σύνολο από APIs που καθιστούν εφικτή την εφαρμογή δικτυακών λειτουργιών όπως είναι η δρομολόγηση, η ασφάλεια, ο έλεγχος πρόσβασης, η διαχείριση του εύρους ζώνης, η ποιότητα υπηρεσιών, η αποθηκευτική και επεξεργαστική βελτίωση, η χρήση ενέργειας, το traffic engineering κ.α.

Έτσι, με την χρήση της SDN αρχιτεκτονικής είναι ευκολότερο να ορίσουμε συνεπείς πολιτικές κατά μήκος ενός ενσύρματου και ασύρματου δικτύου. Με την εφαρμογή ανοιχτών APIs μεταξύ του επιπέδου ελέγχου και του επιπέδου των εφαρμογών, οι εφαρμογές λειτουργούν σαν μία αφαίρεση του δικτύου, συνεπώς μπορούν να εκμεταλλευτούν τις δικτυακές υπηρεσίες και ικανότητες χωρίς να είναι προσκολλημένες στις λεπτομέρειες της εκτέλεσης.

1.5 Περιπτώσεις Χρήσης του SDN

Ο ONF αποτελείται από εταιρίες λογισμικού, από παρόχους, από κατασκευαστές προϊόντων δικτύωσης και προγραμματιστές οι οποίοι όλοι μαζί συμβάλουν στον να υιοθετηθεί όσο το δυνατόν γρηγορότερα η χρήση της SDN αρχιτεκτονικής.

Campus: Τα τελευταία χρόνια τα τμήματα IT των επιχειρήσεων βρίσκονται υπό τρομερή πίεση, ο κύριος λόγος είναι οι απαιτήσεις των τελικών χρηστών για πρόσβαση σε εφαρμογές και δεδομένα από οπουδήποτε και οποτεδήποτε. Οι κινητές συσκευές όπως είναι τα smart phones, τα tablets αυξάνονται στο περιβάλλον ενός δικτύου Campus, οι χρήστες αποθηκεύουν και ταυτόχρονα έχουν πρόσβαση σε ευαίσθητα δεδομένα αυτών των συσκευών τα οποία ανήκουν στον χρήστη και όχι στην επιχείρηση. Ένα Campus δίκτυο πρέπει όχι μόνο να είναι ασφαλές, κλιμακωτό (scalable), και διαχειρίσιμο, πρέπει ταυτόχρονα να διατηρεί την απομόνωση μεταξύ μίας ποικιλίας χρηστών, εφαρμογών, υπηρεσιών, συσκευών αλλά και τεχνολογιών πρόσβασης.

Cloud: Είτε πρόκειται για υβριδικό είτε για ιδιωτικό cloud είναι αρκετά τα πλεονεκτήματα που έχουν οι επιχειρήσεις από την υλοποίηση της SDN αρχιτεκτονικής. Ένα λογικό κεντρικό επίπεδο ελέγχου μπορεί να παρέχει μια πιο κατανοητή άποψη για το data center, τους πόρους του cloud και των δικτύων πρόσβασης. Συνεπώς, εξασφαλίζεται ότι σε περιπτώσεις που υπάρχουν cloud-bursts , οι ροές της κίνησης θα κατευθύνονται σε data centers, τα οποία διαθέτουν επαρκείς πόρους για την εξυπηρέτηση των χρηστών, καθώς και σε συνδέσεις που παρέχουν επαρκές εύρος ζώνης.

Data Center: Μία από τις σημαντικότερες απαιτήσεις που συναντάμε σε ένα κέντρο δεδομένων (Data Center) είναι η εύρεση τρόπων για την βελτίωση της κλιμάκωσης ώστε να είναι δυνατή η υποστήριξη εκατοντάδων Servers αλλά και των εικονικών μηχανών (Virtual Machines) που τρέχουν σε αυτούς. Ωστόσο, για να επιτευχθεί τέτοιου είδους κλιμάκωση είναι απαραίτητη η χρήση ενός μεγάλου αριθμού Servers. Επίσης, οι πίνακες προώθησης των δικτυακών συσκευών χρειάζεται να είναι αρκετά μεγάλοι οδηγώντας τις επιχειρήσεις σε αγορά ακριβών και εξελιγμένων λύσεων. Ακόμη, η διαχείριση της κίνησης και η επιβολή πολιτικών είναι κρίσιμα ζητήματα από την στιγμή που τα Data Centers αναμένεται να επιτυγχάνουν υψηλά επίπεδα απόδοσης. Επιπλέον, τα σύγχρονα Data Centers έχουν πολλές σχεδιαστικές απαιτήσεις μεταξύ των οποίων είναι, η εύκολη μετακίνηση των εικονικών μηχανών, η αποτελεσματική επικοινωνία μεταξύ των server, η ελάχιστη δυνατή συμφόρηση και η απλή διαμόρφωση των routers και των switches.

Στα παραδοσιακά Data Centers οι παραπάνω απαιτήσεις ικανοποιούνται με προσεκτική σχεδίαση και διαμόρφωση των δικτυακών συσκευών. Στις περισσότερες περιπτώσεις αυτό γίνεται χειροκίνητα ορίζοντας τις προτιμώμενες διαδρομές της κίνησης και τοποθετώντας *middleboxes σε διάφορα σημεία του δικτύου.

Όπως είναι προφανές, η διαδικασία της χειροκίνητης διαμόρφωσης είναι απαιτητική και συνάμα πολύπλοκη, επίσης αρκετές φορές οδηγεί σε σφάλματα ειδικά όσο ένα δίκτυο μεγαλώνει. Ακόμη, είναι δύσκολο για το Data Center να χρησιμοποιήσει όλη του την χωρητικότητα, ο κύριος λόγος είναι ότι δεν μπορεί να προσαρμοστεί στις δυναμικές απαιτήσεις των εφαρμογών. Τα οφέλη που προσφέρει το SDN είναι δικτυακές συσκευές οι οποίες είναι πιο εύκολο να διαχειριστούν και παράλληλα είναι φθηνότερες. Την ίδια στιγμή ο δικτυακός έλεγχος ανατίθεται σε μία κεντρική οντότητα (SDN ελεγκτής). Αυτό επιτρέπει την δυναμική διαχείριση των ροών, την εξισορρόπηση της κίνησης και την διανομή πόρων με ένα τρόπο που ταιριάζει στο προφίλ των εφαρμογών που τρέχουν στο Data Center. Ακόμη, η SDN αρχιτεκτονική διευκολύνει την εικονικοποίηση του δικτύου, κάτι που δίνει την δυνατότητα για υψηλή κλιμάκωση εντός του κέντρου δεδομένων, για αυτοματοποιημένη μετακίνηση των εικονικών μηχανών, για καλύτερη αξιοποίηση του server, για χαμηλότερη χρήση ενέργειας και για βελτιστοποίηση του εύρους ζώνης.

Δίκτυα Κορμού: Για την υποστήριξη της πληθώρας των εφαρμογών που προσφέρει το WAN, η Google διαθέτει δύο δίκτυα κορμού. Το πρώτο μεταφέρει την κίνηση των χρηστών και το δεύτερο μεταφέρει την κίνηση μεταξύ των Data Centers. Το πρόβλημα που αντιμετώπισε η Google ήταν το κόστος ανά bit δεν μειωνόταν, ανάλογα με το μέγεθος του δικτύου. Οι κύριοι λόγοι που οδήγησαν στο συγκεκριμένο ζήτημα είναι, ο τρόπος αλληλεπίδρασης μεταξύ των δικτυακών συσκευών, η χειροκίνητη διαμόρφωση και διαχείριση του δικτύου αλλά και η χρήση μη προτυποποιημένων APIs.

Η λύση της Google στο πρόβλημα, ήταν το B4, στη ουσία πρόκειται για χρήση του SDN στο δίκτυο κορμού WAN της Google. Στην ουσία ένας κεντρικός ελεγκτής ορίζει τις διαδρομές ανάμεσα σε δύο σημεία, ικανοποιώντας με αυτό τον τρόπο τους περιορισμούς χωρητικότητας δικτύου και επιτυγχάνοντας παράλληλα καλύτερη σύγκλιση. Από τα αποτελέσματα φάνηκε να υπάρχει καλύτερη αξιοποίηση του δικτύου, λόγω της καθολικής οπτικής και της σχεδιασμένης τοποθέτησης πόρων κατέστη ευκολότερη η αξιολόγηση του δικτύου.

Σημεία Ανταλλαγής Του Διαδικτύου: Ένα σημείο ανταλλαγής του διαδικτύου (Internet Exchange Point) είναι ένα δημόσιο σημείο σύνδεσης το οποίο παρέχει πρόσβαση στο διαδίκτυο. Επίσης είναι εκείνη η τοποθεσία του δικτύου όπου γίνεται η ανταλλαγή κίνησης μεταξύ των παρόχων (ISPs). Για την δρομολόγηση της κίνησης μεταξύ τους, οι πάροχοι χρησιμοποιούν σαν πρωτόκολλο το Border Gateway Protocol (BGP) το οποίο εμπεριέχει κάποιους περιορισμούς. Για παράδειγμα η δυνατότητα να δρομολογεί κίνηση μόνο με βάση το πρόθεμα (Prefix) της IP διεύθυνσης προορισμού. Επίσης δεν δύναται να επηρεάζει μία διαδρομή κίνησης από άκρο σε άκρο, ούτε μπορεί να συμμετέχει σε εφαρμογές ειδικές για ανταλλαγή κίνησης. Η χρήση SDN ελεγκτών πραγματοποιεί απευθείας εξισορρόπηση φορτίου επανεγγράφοντας κεφαλίδες πακέτων.

**Middleboxes: Τοίχος Προστασίας (Firewall), Σύστημα Ανίχνευσης Εισβολής (Intrusion Detection System), Μεταφραστής Διεύθυνσης Δικτύου (NAT), Εξισορροπητές Κίνησης (Load Balancer).*

Άλλες Περιπτώσεις Χρήσης Του SDN: Μερικοί ακόμη τομείς εφαρμογής του SDN είναι τα μικρά αλλά και τα οικιακά δίκτυα, όπου η διαχείριση ανατίθεται σε τρίτα μέρη και ο έλεγχος των switch γίνεται απομακρυσμένα. Στα παραπάνω δίκτυα εφαρμόζεται κατανεμημένη παρακολούθηση του δικτύου για τον εντοπισμό σφαλμάτων αλλά και προειδοποιήσεων ασφαλείας. Μία ακόμη εφαρμογή είναι ο δυναμικός έλεγχος με σκοπό την χορήγηση και επικύρωση της ταυτότητας των χρηστών του δικτύου. Μπορεί να επιτευχθεί αδιάλειπτη κινητικότητα, επιτρέποντας στις συσκευές να συνδέονται σε πολλαπλά δίκτυα κάθε φορά κάτι που υπόσχεται καλύτερη συνδεσιμότητα.

1.6 SDN Cloud Και OpenStack

Η νεφουπολογιστική (Cloud computing) έχει αναπτυχθεί αρκετά γρήγορα το τελευταίο διάστημα. Για τον κόσμο του IT, αυτό σημαίνει πως η υπάρχουσα υποδομή αλλάζει σε σχέση με τον παραδοσιακό τρόπο λειτουργίας σε μία πιο δυναμική και cloud προσέγγιση. Η αρχή έγινε με την εικονικοποίηση των server. Επίσης, ένας ακόμη παράγοντας που έχει οδηγήσει την υιοθέτηση των cloud- εφαρμογών είναι διαφορετικές ομάδες οι οποίες χρειάζεται να συνεργαστούν σε πραγματικό χρόνο. Έτσι, όταν οι servers μπορούν να μετακινηθούν από φυσικές σε εικονικές μηχανές αυτό έχει σαν συνέπεια την πραγματοποίηση αλλαγών και στον τομέα της δικτύωσης

Η εικονικοποίηση των server προσφέρει στην υποδομή του cloud ένα μέρος της ευελιξίας που απαιτείται. Από την άλλη πλευρά, το κομμάτι της δικτύωσης πρέπει να είναι δυναμικό και να προσφέρει όσο το δυνατόν μεγαλύτερη κλιμάκωση. Δύο τεχνολογίες φαίνεται να το κατορθώνουν, η πρώτη είναι αυτή στην οποία επικεντρώνεται η συγκεκριμένη πτυχιακή δηλαδή τα δίκτυα SDN. Στην δεύτερη γίνεται αναφορά στην ενότητα **1.7** και είναι το NFV (Network Functions Virtualization). Από την πλευρά της πλατφόρμας του cloud, το OpenStack είναι ίσως η πιο δημοφιλής επιλογή.

Το OpenStack είναι μία πλατφόρμα ανοιχτού κώδικα που επιτρέπει την δημιουργία του λεγόμενου ‘’Υποδομή σαν υπηρεσία’’ (Infrastructure as a Service). Στη συγκεκριμένη αρχιτεκτονική υπάρχει ένα επίπεδο ελέγχου το οποίο ελέγχει όλα τα εικονικά επίπεδα και παρέχει πρόσβαση στους πόρους ανεξάρτητα από την υποκείμενη τεχνολογία hypervisor (Kernel Virtual Machine, Vmware). Το OpenStack συνδυάζει εργαλεία ανοιχτού κώδικα, τα οποία είναι γνωστά σαν projects, για την πραγματοποίηση λύσεων σε αντίστοιχα ζητήματα. Κάποια από τα project με τα οποία ασχολείται το OpenStack περιλαμβάνουν τους τομείς της υπολογιστικής (compute), της δικτύωσης (networking) και της αποθήκευσης (storage).

Το OpenStack χρησιμοποιεί μία αρθρωτή αρχιτεκτονική, έτσι υπάρχουν μέρη της υποδομής τα οποία είναι εύκολο να μετακινηθούν. Η συγκεκριμένη αρχιτεκτονική επιτρέπει στον χρήστη να εγκαθιστά μόνο ότι είναι απαραίτητο, γλιτώνοντας με αυτό τον τρόπο αρκετά έξοδα. Σε ένα datacenter υπάρχει μία δεξαμενή από πόρους είτε είναι υπολογιστικής, είτε δικτύωσης είτε αποθήκευσης, οι οποίοι ελέγχονται από το OpenStack. Το OpenStack επιτρέπει στους χρήστες να χειρίζονται τους πόρους μέσω μίας web διεπαφής.

Το OpenStack αποτελείται από διάφορα ανεξάρτητα μέρη τα οποία ονομάζονται OpenStack υπηρεσίες. Τα διάφορα μέρη πιστοποιούν την ταυτότητα τους μέσω μίας υπηρεσίας ταυτοποίησης (Identity Service). Οι υπηρεσίες που χρησιμοποιεί το OpenStack επικοινωνούν μεταξύ τους κάνοντας χρήση APIs (Application Programming Interface). Αυτό βέβαια δεν ισχύει στην περίπτωση όπου πρέπει να δοθούν εντολές από τον διαχειριστή. Εσωτερικά, κάθε υπηρεσία του OpenStack αποτελείται από αρκετές διαδικασίες. Όλες οι υπηρεσίες έχουν τουλάχιστον μία API διαδικασία ακρόασης, προεπεξεργασίας καθώς και μετάδοσης API αιτημάτων σε άλλα μέρη της υπηρεσίας.

Ένα από τα project με το οποίο ασχολείται το OpenStack είναι το Neutron, το οποίο αποσκοπεί στην παροχή του ‘’Δικτύου σαν υπηρεσία’’ (Network as a Service). Το Neutron επιτρέπει στους χρήστες (tenants) να δημιουργήσουν εικονικές δικτυακές τοπολογίες που περιλαμβάνουν υπηρεσίες όπως είναι το τείχος προστασίας (firewall), οι εξισορροπητές φορτίου (load balancer) τα εικονικά ιδιωτικά δίκτυα (Virtual Private Networks) κ.α. Το Neutron είναι ίσως από τα πιο περίπλοκα project του OpenStack δεδομένου ότι βασίζεται σε βασικές έννοιες δικτύωσης.

Το Neutron παρέχει δίκτυα, υποδίκτυα, routers σαν αφαιρέσεις αντικειμένων. Υπάρχουν δύο τύποι δικτύων που εφαρμόζονται στο Neutron.Αυτά είναι: τα δίκτυα των παρόχων και τα δίκτυα των χρηστών (tenants). Η διαφορά έγκειται στο ποιος τα διαχειρίζεται. Έτσι, οι χρήστες είναι υπεύθυνοι για την δημιουργία και την διαμόρφωση των δικτύων τους, ώστε να υπάρχει συνδεσιμότητα εντός άλλων project και οι διαχειριστές του OpenStack είναι υπεύθυνοι για την δημιουργία ενός δικτύου παρόχου το οποίο χρησιμοποιούν οι χρήστες.

Εξορισμού τα δίκτυα των χρηστών είναι πλήρως απομονωμένα από άλλα projects. Οι χρήστες έχουν τον πλήρη έλεγχο της τοπολογίας του δικτύου. Οι εικονικοί routers είναι υπεύθυνοι για την δρομολόγηση της κίνησης μεταξύ των δικτύων που ανήκουν στο ίδιο project ή την δρομολόγηση της κίνησης σε εξωτερικά δίκτυα. Μέσα σε ένα project το Neutron παρέχει υπηρεσίες όπως είναι το DHCP (Dynamic Host Configuration Protocol), το DNS (Domain Name System), υπηρεσίες τείχους προστασίας και εξισορρόπησης φορτίου αλλά και την διαμόρφωση ενός VPN δικτύου. Το OpenStack υποστηρίζει τέσσερις τύπους δικτυακής απομόνωσης και τεχνολογιών επικάλυψης: flat,VLAN, GRE (Generic Router Encapsulation) και VXLAN. Από την άλλη πλευρά τα δίκτυα των παρόχων αντιστοιχούν σε υπάρχοντα φυσικά δίκτυα στο datacenter.

Η λύση που προσφέρει το SDN είναι η εφαρμογή δικών του agents Layer 2 και Layer 3 μεταξύ των κόμβων του OpenStack, ώστε να εξαλειφθεί το ζήτημα της συμφόρησης που αντιμετωπίζει ένας agent Layer 3 του Neutron. Οι ελεγκτές του SDN συγκεντρώνουν την διαχείριση των φυσικών και εικονικών δικτύων, έτσι ώστε να διευκολύνουν την διαχείριση και την παρακολούθηση των εργασιών. Επιπλέον, το SDN παρέχει μέσω του κεντρικού επιπέδου ελέγχου αρκετές δικτυακές αφαιρέσεις κάτι που το καθιστά ιδανικό για το OpenStack. Η ενσωμάτωση του SDN στο OpenStack μπορεί να οδηγήσει σε καλύτερη αφαίρεση δικτύων αλλά και πιο αποδοτικό προγραμματισμό των APIs. Η κεντρική διαχείριση της SDN αρχιτεκτονικής ωφελεί την υποδομή του cloud που βασίζεται σε πολλούς κατασκευαστές όπως το OpenStack.

1.7 SDN και NFV

Μία τεχνολογία που τείνει να συγχέεται τα τελευταία χρόνια με το SDN είναι η εικονικοποίηση των λειτουργιών του δικτύου (Network Functions Virtualization, NFV), ίσως λόγω του ότι οι δύο τεχνολογίες συνδέονται αρκετά. Το NFV δίνει την δυνατότητα στον εκάστοτε διαχειριστή οποιουδήποτε δικτύου να πραγματοποιεί την εικονικοποίηση συγκεκριμένων δικτυακών λειτουργιών, όπως είναι για παράδειγμα η μετάφραση της διεύθυνσης δικτύου (Network Address Translation, NAT), η υπηρεσία ονόματος χώρου (Domain Name Service, DNS) και το caching. Με την εισαγωγή της εικονικοποίησης, είναι εφικτό αυτές οι λειτουργίες να παρέχονται από ένα Server γενικού σκοπού ή από ένα switch, αντί να χρησιμοποιούνται δικτυακές συσκευές ειδικά για αυτό τον σκοπό. Η συγκεκριμένη προσέγγιση μειώνει το κόστος λειτουργίας αλλά και το κόστος συντήρησης και ανάθεσης. Ο κύριος λόγος είναι ότι οι επιχειρήσεις αλλά και οι διαχειριστές δεν χρειάζεται να βασίζονται σε λύσεις υλικού (hardware) οι οποίες παρέχονται από τρίτους. Επίσης, με την χρήση του NFV γίνεται ευκολότερη η διαχείριση του δικτύου από την στιγμή που είναι ευκολότερο πλέον να τροποποιήσουμε τις υπάρχουσες υπηρεσίες ή να εισάγουμε νέες με σκοπό την αντιμετώπιση των απαιτήσεων. Η αφαίρεση των δικτυακών λειτουργιών από το υπάρχον υλικό, σχετίζεται αρκετά με την αφαίρεση του επιπέδου ελέγχου από το επίπεδο των δεδομένων που υποστηρίζεται από το SDN, για τον λόγο αυτό η διάκριση των δύο τεχνολογιών μπορεί να μοιάζει κάπως αόριστη.

Είναι σημαντικό να γίνει κατανοητό ότι το SDN και το NFV είναι στενά συνδεδεμένα, αλλά το NFV είναι συμπληρωματικό του SDN και η λειτουργία του δεν βασίζεται σε αυτό αλλά και το αντίστροφο. Για παράδειγμα, οι λειτουργίες ελέγχου του SDN μπορούν να εφαρμοστούν σαν εικονικές λειτουργίες της NFV τεχνολογίας. Από την άλλη πλευρά είναι δυνατό ένα NFV σύστημα ενορχήστρωσης να ελέγχει την τρόπο που προωθούν την κίνηση φυσικά switch μέσω του SDN. Ωστόσο, καμία τεχνολογία δεν είναι απαραίτητη για την λειτουργία της άλλης, αλλά και οι δύο μπορούν να ωφεληθούν από τα πλεονεκτήματα που και οι δύο προσφέρουν.

Κεφάλαιο 2

2.1 OpenFlow Πρωτόκολλο

Από τα πρώτα χρόνια δημιουργίας της SDN τεχνολογίας, αρκετοί άνθρωποι έχουν την τάση να το συγχέουν με το OpenFlow πρωτόκολλο και να θεωρούν ότι είναι το ίδιο πράγμα. Στην πραγματικότητα το OpenFlow **[3]** είναι ένα υποσύνολο τεχνολογιών που περιλαμβάνει το SDN. Το OpenFlow ορίζει τόσο το πρωτόκολλο επικοινωνίας μεταξύ του επιπέδου των δεδομένων με το επίπεδο ελέγχου του SDN, όσο και μέρος της συμπεριφοράς του επιπέδου των δεδομένων. Το OpenFlow ορίζει τις λειτουργίες μέσω των οποίων μπορούμε να διαχειριστούμε τα switch από ένα κεντρικό σημείο (ελεγκτής).

Είναι το πρώτο πρωτόκολλο επικοινωνίας που επιτρέπει στο επίπεδο ελέγχου (*control plane*) του δικτύου να ορίσει κανόνες προώθησης, με τους οποίους μπορούμε να χειριστούμε τις συσκευές. Με την χρήση του έχουμε άμεση πρόσβαση και διαχείριση στο επίπεδο προώθησης των router και των switch. Το OpenFlow εφαρμόζεται και στα δύο μέρη της σύνδεσης (δηλαδή μεταξύ της δικτυακής συσκευής και του SDN control software) και χρησιμοποιεί την έννοια των ροών για τον προσδιορισμό της κίνησης βάση προκαθορισμένων κανόνων αντιστοίχισης που μπορούν, είτε στατικά, είτε δυναμικά να προγραμματιστούν από τον SDN ελεγκτή. Επίσης το IT μπορεί να ορίσει πως η κίνηση θα περνάει από τις συσκευές βάση παραμέτρων όπως είναι ο τρόπος χρήσης, η εφαρμογή, οι πόροι του IT κ.α.

Το συγκεκριμένο πρωτόκολλο δημιουργήθηκε από το πανεπιστήμιο του Stanford στις Η.Π.Α και είναι έργο ανοιχτού κώδικα. Με βάση την SDN αρχιτεκτονική και το OpenFlow πρωτόκολλο οι δικτυακές συσκευές μετατρέπονται σε πλήρως προγραμματιζόμενα στοιχεία προώθησης. Έως τώρα το OpenFlow, είναι το μοναδικό μη ιδιόκτητο πρωτόκολλο που χρησιμοποιείται για τον προγραμματισμό των switches. Αυτό το κεφάλαιο επικεντρώνεται μόνο στο OpenFlow και τις συμπεριφορές που ορίζονται από αυτό, και όχι σε άλλες ανταγωνιστικές εναλλακτικές λύσεις.

Το OpenFlow έχει σχεδιαστεί ώστε να λειτουργεί με το υπάρχον υλικό (hardware) διαφορετικών κατασκευαστών. Στην ουσία δεν είναι απαραίτητη η χρήση εξειδικευμένου υλικού για την λειτουργία του. Επίσης, υπάρχει ένας αριθμός από κατασκευαστές υλικού, οι οποίοι προσφέρουν συσκευές οι οποίες λειτουργούν με την χρήση του OpenFlow μόνο αλλά και την χρήση του πρωτοκόλλου παράλληλα με το λογισμικό του εκάστοτε κατασκευαστή. Η παράλληλη χρήση επιτυγχάνεται εκχωρώντας κάποιες θύρες για το OpenFlow και παρέχοντας τις υπόλοιπες στο λογισμικό του κατασκευαστή. Στον παρακάτω πίνακα παρουσιάζονται διάφορα μοντέλα switch τα οποία υποστηρίζουν το OpenFlow.

Κατασκευαστής	Μοντέλο Switch	Έκδοση
HP	8200ZL,6600,6200 5400,3500	V1.0
Brocase	Netlron CES 200 Series	V1.0
IBM	RackSwitch G8264	V1.0
Juniper	Junos MX Series	V1.0
Pronto	3290,3780	V1.0
Nec	PF5240, PF5820	V1.0
Pica	P-3290,P-3295,P-3780	V1.2

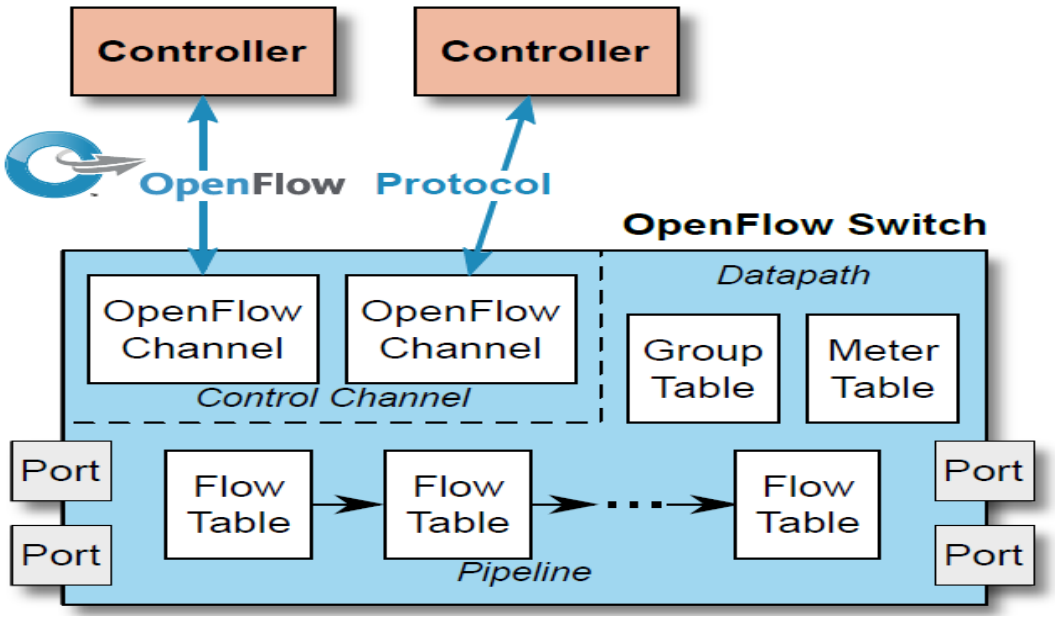
Πίνακας 1: Switch τα οποία υποστηρίζουν την λειτουργία του OpenFlow.

Επίσης, εκτός από τα switch τα οποία είναι συμβατά με το OpenFlow, στον πίνακα 2 παρουσιάζονται οι ελεγκτές οι οποίοι είναι συμβατοί με το OpenFlow.

Ελεγκτής	Υλοποίηση	Προγραμματιστής
POX	Python	Nicira
Beacon	Java	Stanford
FlowVisor	C	Stanford/Nicira
NOX	C	Nicira
Ruy	Python	Ntt
Floodlight	Java	BigSwitch

Πίνακας 2: Ελεγκτές συμβατοί με το OpenFlow.

2.2 OpenFlow Αρχιτεκτονική



Εικόνα 5: Αρχιτεκτονική του OpenFlow.

Ένα OpenFlow-switch υποστηρίζει OpenFlow client λογισμικό και (control plane software) επικοινωνεί με έναν OpenFlow ελεγκτή χρησιμοποιώντας το OpenFlow πρωτόκολλο. Ο ελεγκτής τρέχει σε ένα server ή σε μία φάρμα από servers. Επίσης οι OpenFlow-routers υποστηρίζουν την «αφαίρεση» του πίνακα ροής τον οποίο διαχειρίζεται ο OpenFlow ελεγκτής. Ο πίνακας ροής (flow table) αποτελείται από καταχωρήσεις ροών (flow entries), κάθε καταχώρηση αναπαριστά μία ροή (όπως είναι ένα πακέτο με μία δεδομένη MAC διεύθυνση, VLAN id, IP διεύθυνση, ή συγκεκριμένες TCP/UDP θύρες κ.ο.κ). Ο πίνακας ροής είναι ταξινομημένος βάση προτεραιότητας ροών (flow priority), η προτεραιότητα μίας ροής ορίζεται από τον ελεγκτή. Έτσι, οι καταχωρήσεις με την υψηλότερη προτεραιότητα βρίσκονται πιο ψηλά σε ένα πίνακα ροής.

Ένα OpenFlow-switch μπορεί να είναι οποιαδήποτε συσκευή προώθησης πακέτων (δρομολογητής-router, μεταγωγέας switch) και αποτελείται από έναν ή περισσότερους πίνακες ροών οι οποίοι χρησιμοποιούνται για την επεξεργασία και την προώθηση των πακέτων. Το OpenFlow-switch διαθέτει επίσης και έναν πίνακα ομάδας ο οποίος χρησιμοποιείται και αυτός για την αντιστοίχιση πακέτων αλλά και για προώθηση μέσω ενός ή περισσοτέρων καναλιών σε ένα εξωτερικό ελεγκτή. Το switch επικοινωνεί με τον ελεγκτή και ο ελεγκτής χειρίζεται το switch μέσω του OpenFlow switch πρωτοκόλλου.

Χρησιμοποιώντας το OpenFlow πρωτόκολλο ο ελεγκτής μπορεί να προσθέσει, να διαμορφώσει και να διαγράψει καταχωρήσεις από τον πίνακα ροών, τόσο αντιδραστικά (ως απάντηση σε πακέτα) όσο και προληπτικά, κάνοντας χρήση των στατιστικών στοιχείων που λαμβάνει για διαφορετικές θύρες και ροές. Κάθε πίνακας ροής, περιλαμβάνει μία σειρά από καταχωρήσεις ροών και κάθε καταχώρηση ροής αποτελείται, από πεδία αντιστοίχισης, από μετρητές και ένα σύνολο εντολών που εφαρμόζεται στην περίπτωση που το πακέτο ταιριάζει σε μία καταχώρηση.

Η αντιστοίχιση ξεκινάει στον πρώτο πίνακα ροής και μπορεί να συνεχιστεί και σε άλλους πίνακες. Οι καταχωρήσεις ροής αντιστοιχίζουν τα πακέτα με σειρά προτεραιότητας με την πρώτη καταχώρηση αντιστοίχησης να χρησιμοποιείται. Εάν τα πεδία ενός πακέτου ταιριάζουν με αυτά μίας καταχώρησης , οι εντολές που αφορούν την καταχώρηση ροής εκτελούνται. Εάν δεν υπάρξει καμία αντιστοίχιση στον πίνακα ροής το αποτέλεσμα εξαρτάται από την διαμόρφωση του table-miss πίνακα, για παράδειγμα ένα πακέτο μπορεί να προωθηθεί στους ελεγκτές μέσω του OpenFlow καναλιού, να απορριφθεί ή να συνεχίσει στον επόμενο πίνακα ροής.

Οι εντολές που αφορούν κάθε καταχώρηση είτε περιλαμβάνουν δράσεις (actions) είτε διαμόρφωση της διασωλήνωσης (modify pipeline). Οι δράσεις περιλαμβάνουν εντολές που περιγράφουν την προώθηση του πακέτου, την διαμόρφωση του και την επεξεργασία του πίνακα ομάδας. Οι εντολές επεξεργασίας της διασωλήνωσης επιτρέπουν στα πακέτα να σταλθούν σε μεταγενέστερους πίνακες για περαιτέρω επεξεργασία αλλά και την μεταφορά πληροφορίας μεταξύ των πινάκων σε μορφή μεταδεδομένων. Η διαδικασία τερματίζει όταν οι εντολές που αφορούν μία καταχώρηση δεν προσδιορίζουν ένα νέο πίνακα, στο σημείο αυτό το πακέτο διαμορφώνεται και προωθείται.

Το switch κάνοντας χρήση της εκάστοτε καταχώρησης προωθεί τα πακέτα σε μία θύρα εξόδου (out port) από μία θύρα εισόδου (in port). Η θύρα ενδέχεται να είναι μία φυσική θύρα του switch ή μία λογική θύρα ή μία δεσμευμένη θύρα. Εν κατακλείδι, ένα OpenFlow-switch πρέπει να υποστηρίζει τρεις τύπους θυρών:λογικές, φυσικές και δεσμευμένες θύρες.

Οι OpenFlow φυσικές θύρες ορίζονται από το switch και αντιπροσωπεύουν πραγματικές διεπαφές υλικού (hardware interface), για παράδειγμα σε ένα Ethernet switch οι φυσικές θύρες αντιστοιχούν μία προς μία στις Ethernet διεπαφές του switch. Οι λογικές θύρες ορίζονται από το switch και δεν αντιπροσωπεύουν μία άμεση διεπαφή υλικού, ορίζονται στο switch χρησιμοποιώντας μη-OpenFlow μεθόδους, για παράδειγμα link aggregation groups, tunnels και loopback interfaces. Οι δεσμευμένες θύρες ορίζουν γενικές δράσεις προώθησης που τρέχουν στο switch και ορίζονται από την συσκευή για αυτό το σκοπό, την χρήση τους για εφαρμογή του πρωτοκόλλου OpenFlow χρησιμοποιώντας συγκεκριμένες λειτουργίες προώθησης, όπως αποστολή πακέτων στον ελεγκτή, μαζική προώθηση σε όλες τις συσκευές (flooding) ή και χρήση μη-OpenFlow μεθόδων προώθησης όπως λειτουργεί εξ ορισμού (by default) ο δρομολογητής.

Εκτός από την επεξεργασία κάθε πακέτου ξεχωριστά, κάποιες καταχωρήσεις περιλαμβάνουν δράσεις οι οποίες ενδέχεται να κατευθύνουν τα πακέτα σε μία ομάδα (group). Το switch μπορεί να χρησιμοποιήσει τον πίνακα ομάδας (group table) για πιο μαζική επεξεργασία της κίνησης. Μία καταχώρηση ροής μπορεί να αντιστοιχεί σε μία δράση του πίνακα ομάδας (group table action) για πακέτα που αντιστοιχούν στην εν λόγω καταχώρηση. Οι δράσεις που επιτελούνται με την χρήση του πίνακα ομάδας και των καταχωρήσεων ομάδας ενδέχεται να είναι πολύπλοκες: για παράδειγμα η πολυδιόδευση πακέτων (multipath forwarding), η γρήγορη αναδρομολόγηση (fast reroute) ή η συσσωμάτωση ζεύξεων (link aggregation).

2.3 Επεξεργασία Διασωλήνωσης

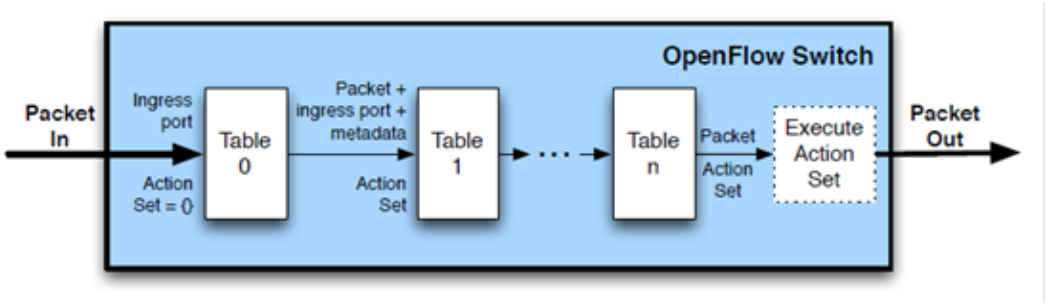
Οι πίνακες ροής των switch βρίσκονται σε διασωληνωμένη μορφή και ανάλογα με την μορφή της διασωλήνωσης διακρίνονται σε δύο κατηγορίες: OpenFlow-only και OpenFlow-υβριδικά.Τα OpenFlow-only switch υποστηρίζουν λειτουργίες OpenFlow μόνο, στα συγκεκριμένα switch όλα τα πακέτα επεξεργάζονται από τον OpenFlow αγωγό διότι δεν γίνεται να τα επεξεργαστούμε με άλλο τρόπο.

OpenFlow-υβριδικά : Αυτά τα switch υποστηρίζουν OpenFlow λειτουργίες και κανονικές Ethernet switching λειτουργίες, για παράδειγμα επιπέδου δύο (Layer 2) Ethernet switching, VLAN απομόνωση, επιπέδου τρία (Layer 3) δρομολόγηση, ACL, QoS επεξεργασία. Οι συγκεκριμένες συσκευές παρέχουν ένα μηχανισμό ταξινόμησης, ο οποίος ανακατευθύνει τα πακέτα είτε στη διασωλήνωση του OpenFlow είτε στη κανονική διασωλήνωση. Για παράδειγμα, ένα switch μπορεί να χρησιμοποιήσει την VLAN υπογραφή ή την θύρα εισόδου ενός πακέτου για να καθορίσει εάν θα επεξεργαστεί το πακέτο και ποια διασωλήνωση θα χρησιμοποιήσει, ή για να κατευθύνει το πακέτο στην OpenFlow διασωλήνωση.

OpenFlow-only : Αυτά τα switch εμπεριέχουν έναν ή περισσότερους πίνακες ροής, κάθε πίνακας περιλαμβάνει πολλαπλές καταχωρήσεις. Η επεξεργασία της διασωλήνωσης ορίζει πως τα πακέτα αλληλεπιδρούν με αυτούς τους πίνακες.

Τα OpenFlow switch απαιτείται να έχουν τουλάχιστον ένα πίνακα ροής , την στιγμή που ένα switch με ένα μόνο πίνακα ροής είναι έγκυρο απλουστεύοντας την επεξεργασία της διασωλήνωσης.

Για την διαδοχική σύγκριση των πινάκων ροής το OpenFlow ακολουθεί διασωληνωμένη λογική (pipeline logic), δημιουργείται έτσι μία σταθερή ροή πακέτων μεταξύ των πινάκων ροής όπως φαίνεται στο παρακάτω σχήμα:



Εικόνα 6: *Ροή ενός πακέτου μέσω του αγωγού επεξεργασίας*

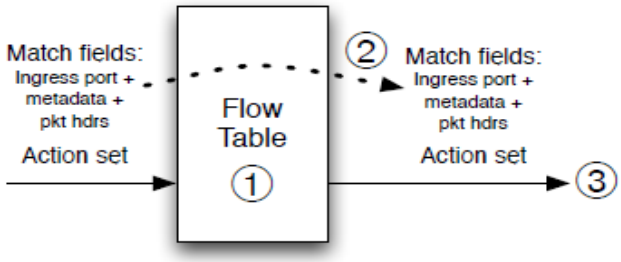
Οι πίνακες ροής ενός OpenFlow-switch αριθμούνται ακολουθιακά ξεκινώντας από το μηδέν (0). Η επεξεργασία της διασωλήνωσης ξεκινάει από τον πρώτο πίνακα ροής: Το πακέτο ταυτοποιείται πρώτα με τις καταχωρήσεις ροών του πίνακα ροής μηδέν (0). Επίσης, μπορούν να χρησιμοποιηθούν και άλλοι πίνακες ροής, αυτό εξαρτάται από το αποτέλεσμα της αντιστοίχισης στον πρώτο πίνακα ροής.

Κατά την επεξεργασία του πακέτου από τον πρώτο πίνακα ροής, γίνεται σύγκριση των δεδομένων της κεφαλίδας του πακέτου, της θύρας εισόδου του πακέτου και τυχόν μεταδεδομένων διαδοχικά με τις καταχωρήσεις ροών του πρώτου πίνακα ροής. Η αντιστοίχιση πραγματοποιείται με σειρά προτεραιότητας, δηλαδή η πρώτη επιτυχής αντιστοίχιση καταχώρησης είναι αυτή που ακολουθείται, με τις επόμενες καταχωρήσεις στους υπόλοιπους πίνακες να χρησιμοποιούνται ανάλογα με τις οδηγίες που υπάρχουν στην υπάρχουσα καταχώρηση.

Μόλις βρεθεί η καταχώρηση που ταιριάζει με το πακέτο, εκτελείται το αντίστοιχο σύνολο εντολών (instruction set) που περιλαμβάνει η καταχώρηση. Οι εντολές αυτές ενδέχεται να ανακατευθύνουν το πακέτο σε άλλη καταχώρηση, σε ένα επόμενο πίνακα ροής όπου η ίδια διαδικασία επαναλαμβάνεται (π.χ Goto-Table εντολή). Στο σημείο αυτό αξίζει να τονίσουμε το γεγονός ότι μία καταχώρηση μπορεί να κατευθύνει ένα πακέτο σε πίνακα ροής με αύξοντα αριθμό μεγαλύτερο από τον υπάρχων πίνακα (στο οποίο υπάρχει η καταχώρηση). Έτσι, η αρχιτεκτονική της διασωλήνωσης μπορεί να λειτουργεί μόνο με προώθηση προς τα εμπρός και όχι προς τα πίσω.

Όταν το πακέτο φτάσει στο τέλος της διασωλήνωσης (στον τελευταίο πίνακα ροής) και δεν υφίσταται άλλη εντολή προώθησης, τότε το πακέτο επεξεργάζεται με τις εντολές που έχει η συγκεκριμένη καταχώρηση και προωθείται στην ανάλογη θύρα εξόδου.

Γενικότερα κατά την εκτέλεση οποιασδήποτε εντολής σε ένα πίνακα ροής η επεξεργασία γίνεται όπως φαίνεται στην παρακάτω εικόνα:



Εικόνα 7: Επεξεργασία του πακέτου σε ένα πίνακα ροής

i) Εύρεση της καταχώρησης με την υψηλότερη προτεραιότητα.

ii) Εκτέλεση εντολών.

- Τροποποίηση του πακέτου & ενημέρωση των πεδίων αντιστοίχισης.
- Ενημέρωση του συνόλου ενεργειών.
- Ενημέρωση των μεταδεδομένων.

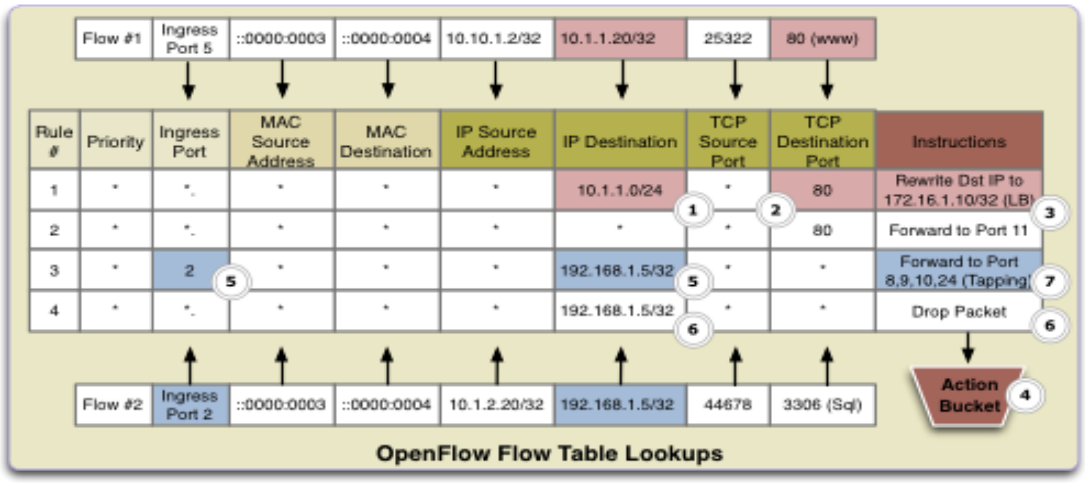
iii) Αποστολή των δεδομένων αντιστοίχισης και του συνόλου ενεργειών στον επόμενο πίνακα.

Το πεδίο των μεταδεδομένων μπορεί να χρησιμοποιηθεί στην περίπτωση που σε έναν πίνακα θέλουμε να ταξινομήσουμε δύο τύπους πακέτων. Αυτοί οι δύο τύποι πακέτων ενδεχομένως να έχουν δύο διαφορετικές τιμές μεταδεδομένων. Έτσι, μπορούμε να κάνουμε χρήση του πεδίου των μεταδεδομένων για να περάσουμε το τύπο του πακέτου (δηλαδή την ταξινόμηση που έγινε στον προηγούμενο πίνακα) στον επόμενο πίνακα.

Εάν το πακέτο δεν ταιριάζει με καμία καταχώρηση του πίνακα αυτό ονομάζεται table-miss. Οι εντολές που υπάρχουν σε μία table-miss καταχώρηση ορίζουν την επεξεργασία τέτοιων πακέτων. Μερικές από τις επιλογές είναι η απόρριψη του πακέτου, η αποστολή σε άλλο πίνακα ή η αποστολή στον ελεγκτή μέσω του καναλιού ελέγχου.

2.4 Πίνακες Ροής Και Καταχωρήσεις

Ένας πίνακας ροής αποτελείται από καταχωρήσεις ροών, όπως φαίνεται και στην εικόνα 8. Τα πεδία κεφαλίδας (header fields) χρησιμοποιούνται σαν κριτήρια αντιστοίχισης ώστε να καθοριστεί εάν ένα εισερχόμενο πακέτο ταιριάζει σε μία καταχώρηση. Σε περίπτωση που υπάρχει αντιστοίχιση του πακέτου σε μία καταχώρηση τότε το πακέτο ανήκει στην συγκεκριμένη ροή.



Εικόνα 8:Πίνακας ροής

Κάθε καταχώρηση ροής περιλαμβάνει

- **Πεδία αντιστοίχισης:** για αντιστοίχιση με τα πακέτα.Αποτελούνται από την θύρα εισόδου και τις κεφαλίδες του πακέτου και προαιρετικά μπορεί να περιλαμβάνουν άλλα πεδία του αγωγού (όπως το metadata πεδίο που ορίζεται από ένα προηγούμενο πίνακα) .
- **Προτεραιότητα:** αντιστοίχιση προτεραιότητας μίας καταχώρησης ροής.
- *** Μετρητές:** Ανανεώνονται όταν πακέτα αντιστοιχούνται.

- **Εντολές :** Για την διαμόρφωση της δέσμης ενεργειών ή του αγωγού επεξεργασίας.
- **Timeouts:** Ορίζουν τον μέγιστο χρονικό όριο πριν το switch αφαιρέσει μία ροή.
- **Cookie:** Αδιαφανής τιμές δεδομένων επιλεγμένες από τον ελεγκτή.Δεν χρησιμοποιούνται κατά την επεξεργασία του πακέτου.
- **Flags:** Οι σημαίες αλλάζουν τον τρόπο διαχείρισης των καταχωρήσεων.

Match Fields		Priority	Counters	Instructions	Timeouts	Cookie	Flags

Πίνακας 3: Βασικά πεδία μιας καταχώρησης ροής.

Μία καταχώρηση αναγνωρίζεται από τα πεδία αντιστοίχισης και προτεραιότητας της. Χρησιμοποιώντας τα δύο αυτά πεδία αναγνωρίζεται μία μοναδική καταχώρηση σε ένα συγκεκριμένο πίνακα ροής. Η καταχώρηση που παραλείπει όλα τα πεδία και έχει προτεραιότητα ίση με το μηδέν ονομάζεται table-miss flow καταχώρηση.

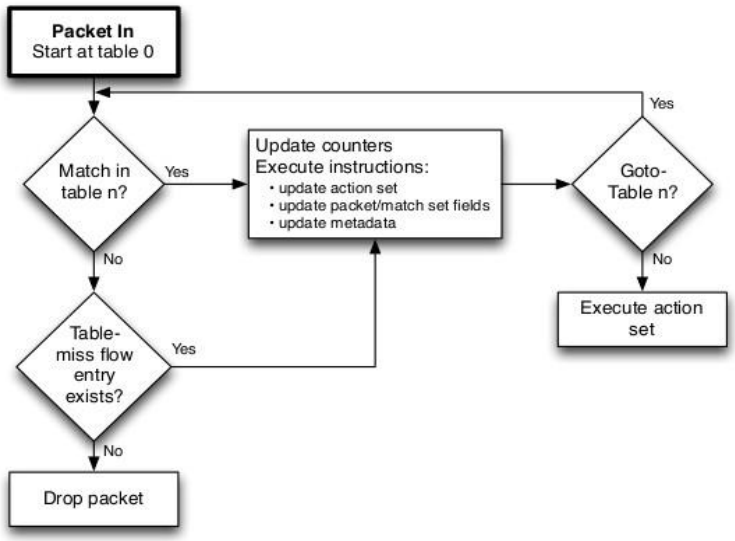
*Μετρητές: Χρησιμοποιείται για την συλλογή στατιστικών για κάθε ροή, όπως είναι ο αριθμός των λαμβανόμενων πακέτων, ο αριθμός των bytes καθώς και η διάρκεια της ροής.

Οι εντολές μίας καταχώρησης ενδέχεται να περιλαμβάνουν δράσεις που πραγματοποιούνται στο πακέτο σε κάποιο σημείο του αγωγού. Να σημειώσουμε πως κάθε πίνακας δεν μπορεί να υποστηρίζει κάθε πεδίο αντιστοίχισης, κάθε εντολή και κάθε δράση. Τα χαρακτηριστικά των αιτημάτων είναι αυτά που επιτρέπουν στον ελεγκτή να καταλάβει τι υποστηρίζει κάθε πίνακας.

2.5 Αντιστοίχιση

Όταν ένα πακέτο καταφθάνει στο OpenFlow-switch από μία εισερχόμενη θύρα (ή σε κάποιες περιπτώσεις από τον ελεγκτή), εκτελείται μία διαδικασία αναζήτησης και ταύτισης των καταχωρήσεων που υπάρχουν στους πίνακες ροής της συσκευής, σε σχέση με τα δεδομένα που φέρει το πακέτο στην επικεφαλίδα του (header). Τα πεδία του πακέτου χρησιμοποιούνται για αντιστοίχιση με τις καταχωρήσεις του πρώτου πίνακα ροής και ανάλογα με την επεξεργασία του αγωγού, το switch ενδέχεται να πραγματοποιήσει αντιστοιχίσεις και σε άλλους πίνακες ροής, όπως φαίνεται και στην εικόνα 9.

Τα πεδία κεφαλίδας του πακέτου που χρησιμοποιούνται για αντιστοιχίσεις πινάκων διαφέρουν ανάλογα με τον τύπο του πακέτου και συνήθως περιλαμβάνουν διάφορα πρωτόκολλα όπως το Ethernet, το IPv4, IPv6, MPLS κ.α. Επίσης αντιστοίχιση πραγματοποιείται στην θύρα εισόδου, στο πεδίο των metadata και σε άλλα πεδία του αγωγού.



Εικόνα 9:Ροή ενός πακέτου σε ένα OpenFlow switch

Ένα πακέτο ταιριάζει σε μία καταχώρηση επιτυχώς όταν οι τιμές στα πεδία της αντιστοίχισης του πακέτου ταυτίζονται ακριβώς με τα πεδία αντιστοίχισης που ορίζονται στην καταχώρηση ροής. Εάν το πεδίο αντιστοίχισης μίας καταχώρησης παραλείπεται (έχοντας τιμή ANY), τότε ταιριάζει με όλες τις πιθανές τιμές στα πεδία της κεφαλίδας ή στα πεδία του αγωγού του πακέτου. Εάν το πεδίο αντιστοίχισης είναι παρών και δεν περιλαμβάνει κάποια μάσκα,τότε ταιριάζει με το αντίστοιχο πεδίο κεφαλίδας ή αγωγού από το πακέτο εφόσον έχει την ίδια τιμή.

Η σάρωση των πεδίων των πακέτων στο OpenFlow βασίζεται στην δομή των πλαισίων όπως ορίζονται από το πρωτόκολλο Ethernet. Λόγω όμως του ότι υπάρχουν πολλά είδη πλαισίωσης

στο Ethernet η αντιστοίχιση από το OpenFlow γίνεται με βάση του τι θεωρείται από το OpenFlow ως το ωφέλιμο φορτίο (payload) του πακέτου. Έτσι μπορεί να εντοπιστεί το πεδίο Ethertype στο πλαίσιο Ethernet και να καθοριστεί ποιο πρωτόκολλο είναι ενθυλακωμένο στο πλαίσιο (συνήθως MPLS, Internet Protocol (IP), ARP) όπως στον πιο κάτω πίνακα.

Preamble	Start of frame delimiter	Προορισμός MAC	Πηγή MAC	Ετικέτα VLAN	Ethertype (Ethernet II) or length (IEEE 802.3)	Payload	32−bit CRC	Interframe gap
7 octets	1 octet	6 octets	6 octets	4 octets	2 octets	42–1500 octets	4 octets	12 octets
64-1522 octets								
72-1530 octets								
84-1542 octets								

Πίνακας 4: Το πλαίσιο του Ethernet.

Για παράδειγμα κατά την λήψη πακέτου με ετικέτα VLAN (virtual local area network) το Ethertype βρίσκεται αμέσως μετά το πεδίο ετικέτα VLAN στο πιο πάνω σχήμα. Εξαίρεση αποτελεί το MPLS (multiprotocol layer switching) που λόγω της δομής του, το OpenFlow δεν μπορεί να προσδιορίσει το πρωτόκολλο που ενθυλακώνεται στο πλαίσιο, οπότε και χρησιμοποιείται η ετικέτα MPLS για την αντιστοίχιση.

Το πακέτο συγκρίνεται με καταχωρήσεις του πίνακα και μόνο η καταχώρηση με την υψηλότερη προτεραιότητα επιλέγεται. Οι μετρητές που σχετίζονται με την επιλεγμένη καταχώρηση ενημερώνονται και το σύνολο των εντολών που περιλαμβάνει η καταχώρηση εκτελούνται. Στην περίπτωση που υπάρχει αντιστοίχιση του πακέτου με πολλαπλές καταχωρήσεις ταυτόχρονα έχοντας ίδιο βαθμό προτεραιότητας η επιλεγμένη καταχώρηση είναι απροσδιόριστη.

2.6 Εντολές

Κάθε καταχώρηση περιλαμβάνει ένα σύνολο εντολών (instruction set) που εκτελούνται όταν το πακέτο ταιριάζει σε μία καταχώρηση. Αυτές οι εντολές έχουν σαν αποτέλεσμα να μεταβάλλουν το πακέτο, την επεξεργασία της διασωλήνωσης και το σύνολο ενεργειών (action set). Ένα switch δεν είναι απαραίτητο να υποστηρίζει όλους τους τύπους εντολών αλλά μόνο όσων έχουν την ένδειξη “Required” παρακάτω. Ο ελεγκτής επίσης μπορεί να ρωτήσει το switch για τους τύπους προαιρετικών εντολών που υποστηρίζει.

- **Meter meter_id *(*Optional*)***: Κατευθύνει το πακέτο σε ένα συγκεκριμένο μετρητή.
- **Apply-actions (Optional)**: Εφαρμόζονται άμεσα οι ενέργειες που σχετίζονται με το πακέτο χωρίς καμία αλλαγή στο σύνολο ενεργειών.
- **Clear-actions (Optional):** Διαγράφονται όλες οι ενέργειες στο σύνολο ενεργειών.
- **Write-actions (Required):** Συγχωνεύει τις καθορισμένες από την καταχώρηση ροής ενέργειες στο σύνολο ενεργειών του πακέτου.
- **Write-metadata (Optional):** Γράφει τα metadata στο αντίστοιχο πεδίο του match field.

- **Goto-Table (Required):** Υποδεικνύει τον επόμενο πίνακα για μετάβαση μέσα στον αγωγό.

Οι μόνοι περιορισμοί είναι ότι, η εντολή Meter εκτελείται πριν από την εντολή Apply-actions, η εντολή Clear-actions εκτελείται πριν από την εντολή Write-actions και η εντολή Goto-Table εκτελείται τελευταία. Το switch θα πρέπει να απορρίπτει μία καταχώρηση όταν δεν μπορεί να εκτελέσει τις εντολές ή ένα μέρος των εντολών που σχετίζονται με την συγκεκριμένη καταχώρηση. Σε αυτή την περίπτωση το switch θα πρέπει να επιστρέφει ένα μήνυμα λάθους σχετικά με το ζήτημα.

2.7 Δράσεις (Actions)

Οι δράσεις ορίζουν την προώθηση και την διαμόρφωση του πακέτου, αλλά και την επεξεργασία του πίνακα ομάδας (group table). Ένα OpenFlow switch μπορεί να υποστηρίζει διάφορες δράσεις. Παρακάτω παρουσιάζονται οι δράσεις που απαιτείται να υποστηρίζει ένα switch:

- Output: Η δράση *Output* προωθεί το πακέτο σε μία συγκεκριμένη θύρα. Ένα OpenFlow switch πρέπει να υποστηρίζει προώθηση πακέτων τόσο σε φυσικές και όσο και σε λογικές θύρες.
- Group: Η δράση *Group* ορίζει την επεξεργασία του πακέτου μέσω του πίνακα ομάδας.
- Drop: Στην ουσία δεν υπάρχει κάποια ρητή δράση για την απόρριψη ενός πακέτου. Αυτό συμβαίνει στην περίπτωση όπου η δέσμη ενεργειών ενός πακέτου δεν έχει ένα *Output action* ή ένα *Group action*, τότε το αποτέλεσμα είναι το πακέτο να απορρίπτεται. Αυτό το αποτέλεσμα μπορεί να προέλθει εάν το σύνολο εντολών είναι κενό ή μετά την εκτέλεση της εντολής Clear-action.

2.8 Σύνολο Ενεργειών (Action Set)

Κάθε πακέτο συνδέεται με ένα σύνολο ενεργειών (action set), το οποίο είναι κενό από προεπιλογή. Μία καταχώρηση μπορεί να τροποποιήσει το σύνολο ενεργειών χρησιμοποιώντας μία Write-action ή μία Clear-action εντολή. Το σύνολο ενεργειών μεταφέρεται μεταξύ των πινάκων ροής. Όταν το σύνολο ενεργειών μιας καταχώρησης δεν περιλαμβάνει την εντολή Goto-Table, η επεξεργασία του αγωγού σταματάει και οι δράσεις στο σύνολο ενεργειών του πακέτου εκτελούνται. Κάθε σύνολο ενεργειών έχει ένα μέγιστο μιας εντολής για κάθε τύπο πακέτου. Αν χρειάζονται πολλές εντολές για τον ίδιο τύπο μπορεί να χρησιμοποιηθεί η εντολή *apply actions*.

2.9 Μετρητές

Οι μετρητές (*counters*) διατηρούνται για κάθε πίνακα ροής (flow table), καταχώρηση (flow entry), θύρα (port), ουρά (queue), ομάδα (group). Μετρητές συμβατοί με το OpenFlow μπορούν να υλοποιηθούν σε λογισμικό και να διατηρηθούν χρησιμοποιώντας υλικό περιορισμένης εμβέλειας. Ένα switch δεν απαιτείται να υποστηρίζει όλους τους τύπους μετρητών, μόνο όσων έχουν την ένδειξη “Required” στον παρακάτω πίνακα. Η διάρκεια αναφέρεται στο χρονικό διάστημα που έχει περάσει από την εγκατάσταση στο switch μίας καταχώρησης ροής, μίας θύρας, μίας ομάδας, μίας ουράς ή ενός μετρητή. Το πεδίο Receive Errors Field είναι το σύνολο όλων των σφαλμάτων λήψης και σύγκρουσης.

Counter	Bits	
Per Flow Table		
Reference count (active entries)	32	Required
Packet Lookups	64	Optional
Packet Matches	64	Optional
Per Flow Entry		
Received Packets	64	Optional
Received Bytes	64	Optional
Duration (seconds)	32	Required
Duration (nanoseconds)	32	Optional
Per Port		
Received Packets	64	Required
Transmitted Packets	64	Required
Received Bytes	64	Optional
Transmitted Bytes	64	Optional
Receive Drops	64	Optional
Transmit Drops	64	Optional
Receive Errors	64	Optional
Transmit Errors	64	Optional
Receive Frame Alignment Errors	64	Optional
Receive Overrun Errors	64	Optional
Receive CRC Errors	64	Optional
Collisions	64	Optional
Duration (seconds)	32	Required
Duration (nanoseconds)	32	Optional
Per Queue		
Transmit Packets	64	Required
Transmit Bytes	64	Optional
Transmit Overrun Errors	64	Optional
Duration (seconds)	32	Required
Duration (nanoseconds)	32	Optional
Per Group		
Reference Count (ow entries)	32	Optional
Packet Count	64	Optional
Byte Count	64	Optional
Duration (seconds)	32	Required
Duration (nanoseconds)	32	Optional
Per Group Bucket		
Packet Count	64	Optional
Byte Count	64	Optional
Per Meter		
Flow Count	32	Optional
Input Packet Count	64	Optional
Input Byte Count	64	Optional
Duration (seconds)	32	Required
Duration (nanoseconds)	32	Optional
Per Meter Band		
In Band Packet Count	64	Optional
In Band Byte Count	64	Optional

Πίνακας 5: Λίστα Μετρητών

2.10 Πίνακες Ομάδας

Ένα ακόμη είδος πινάκων που ορίζει το OpenFlow είναι αυτό των group tables, τα οποία περιέχουν καταχωρήσεις ομάδας (group entries). Οι πίνακες ομάδας μας επιτρέπουν την αποστολή πολλαπλών αντιγράφων του πακέτου ταυτόχρονα εφαρμόζοντας παράλληλα ένα διαφορετικό σύνολο δράσεων για κάθε πακέτο. Η δομή μίας καταχώρησης ομάδας είναι:

Group Identifier	Group Type	Counters	Action Buckets

Πίνακας 6: Βασικά πεδία μιας καταχώρησης στον πίνακα ομαδοποίησης (group table).

Κάθε καταχώρηση ομάδας αναγνωρίζεται από την ταυτότητα της ομάδας και περιλαμβάνει:

- Group identifier: Ένας 32-bit μη ανατεθειμένος ακέραιος ο οποίος μοναδικά ταυτοποιεί την ομάδα στο OpenFlow switch.
- Group type: Ο τύπος της ομάδας, μας δείχνει ποιες εντολές από την ομάδα εντολών θα εκτελεστούν για τα πακέτα που ανήκουν στην καταχώρηση.
- Counters: Μετρητές που ανανεώνονται όταν πακέτα επεξεργάζονται από μία ομάδα.
- Action buckets: Μία διατεταγμένη λίστα εντολών στην οποία κάθε εντολή περιλαμβάνει ένα σύνολο ενεργειών που πρέπει να εκτελεστούν μαζί με τις σχετικές παραμέτρους σε ένα πακέτο το οποίο βρίσκεται υπό επεξεργασία.

Όσον αφορά τα group types, τα switch δεν είναι απαραίτητο να υποστηρίζουν όλους τους τύπους αλλά μόνο όσα έχουν την ένδειξη “required” παρακάτω. Ο ελεγκτής επίσης μπορεί να ρωτήσει το switch για ποιους από τους προαιρετικούς τύπους (optional) υποστηρίζει. Έτσι έχουμε τέσσερις τύπους ομάδας:

1.all (required): Εκτελούνται όλες οι εντολές στις ομάδες εντολών, με το πακέτο να κλωνοποιείται σε αντίγραφα και να επεξεργάζεται από κάθε εντολή στο action bucket. Αυτή η κατηγορία χρησιμοποιείται συνήθως για ευρυεκπομπή (broadcasting) και πολυεκπομπή (multicasting).

2.select (optional): Εκτελείται μόνο μια εντολή από το σύνολο της ομάδας εντολών, βάση των παραμέτρων του πακέτου και των αλγορίθμων επιλογής της εκάστοτε συσκευής. Αυτός ο τύπος ομάδας χρησιμοποιείται για εξισορρόπηση του φορτίου.

3.indirect (required): Εκτελείται μια μόνο προκαθορισμένη εντολή στο group table, πράγμα που επιτρέπει πολλαπλές ροές πακέτων να δείχνουν σε ένα αναγνωριστικό ομάδας (group identifier) και να υποστηρίζεται γρηγορότερη και αποτελεσματικότερη προώθηση ομάδων πακέτων σε συγκεκριμένο προορισμό.

4.fast failover (optional) : Εκτελείται η πρώτη εν ενεργεία ομάδα εντολών, δηλαδή το σύνολο εντολών που είναι συσχετισμένο με μια ενεργή θύρα εξόδου.

Με αυτή την μέθοδο κάθε σύνολο ενεργειών στο action bucket συσχετίζεται με μια θύρα εξόδου για να μπορεί η συσκευή δρομολόγησης σε περίπτωση απώλειας σύνδεσης από μια θύρα εξόδου να ανακατευθύνει τα πακέτα σε εφεδρικές θύρες χωρίς να χρειαστεί ξανά επικοινωνία με τον ελεγκτή.

2.11 OpenFlow Channel

Το OpenFlow κανάλι είναι η διεπαφή (interface) που συνδέει κάθε OpenFlow switch σε ένα OpenFlow ελεγκτή. Μέσω αυτής της διεπαφής ο ελεγκτής διαμορφώνει και χειρίζεται το switch με σκοπό την λήψη μηνυμάτων για διάφορα γεγονότα (events) αλλά και την αποστολή πακέτων. Η διεπαφή μπορεί να διαφέρει ανάλογα με την υλοποίηση του OpenFlow switch, παρόλα αυτά τα μηνύματα που προορίζονται να σταλούν μέσω του OpenFlow καναλιού πρέπει να είναι τυποποιημένα σύμφωνα με το πρωτόκολλο OpenFlow. Γενικά, η επικοινωνία διασφαλίζεται με ασύμμετρη κρυπτογράφηση και χρήση του πρωτοκόλλου TLS, αν και επιτρέπεται μη κρυπτογραφημένη σύνδεση μέσω του TCP πρωτοκόλλου.
Το switch μπορεί να επικοινωνήσει με τον ελεγκτή μέσω αυτών των συνδέσεων χρησιμοποιώντας την IP διεύθυνση του ελεγκτή και θύρα εισόδου που ορίζεται από τον χρήστη. Όταν γίνεται χρήση του TLS μπορεί να χρησιμοποιηθεί εξ ορισμού η TCP θύρα που είναι η 6633.

2.12 OpenFlow Switch Protocol

Το OpenFlow πρωτόκολλο δημιουργήθηκε ώστε να παρέχει ένα τυποποιημένο τρόπο επικοινωνίας μεταξύ ενός OpenFlow-switch και του ελεγκτή που το ελέγχει. Το OpenFlow υποστηρίζει τρεις τύπους μηνυμάτων επικοινωνίας ελεγκτή-στο switch,ασύγχρονα (asynchronous) και συμμετρικά (symmetric) με το κάθε ένα να διαθέτει τις δικές του υποκατηγορίες. Τα μηνύματα από τον ελεγκτή στο switch ή αλλιώς controller-to-switch αρχικοποιούνται από τον ελεγκτή και χρησιμοποιούνται για άμεση διαχείριση ή επιθεώρηση της κατάστασης ενός switch. Τα ασύγχρονα μηνύματα δημιουργούνται από το switch και ο σκοπός τους είναι να ενημερώσουν τον ελεγκτή σχετικά με συμβάντα του δικτύου ή κάποια αλλαγή της κατάστασης του switch. Τα συμμετρικά μηνύματα δημιουργούνται είτε από το switch είτε από τον ελεγκτή χωρίς προηγουμένως να τους έχει ζητηθεί κάτι τέτοιο. Παρακάτω περιγράφονται αναλυτικότερα οι τρεις κατηγορίες μηνυμάτων.

1. **Controller-to-switch**

- *Features*: Ο ελεγκτής στέλνει στο switch ζητώντας τις βασικές ιδιότητες και την ταυτότητα του (features request) και το switch απαντάει με ένα μήνυμα (features reply) που ορίζει τα παραπάνω. Αυτή η διαδικασία πραγματοποιείται κατά την εγκατάσταση του OpenFlow καναλιού.

- *Configuration:* Ο ελεγκτής είναι σε θέση να ορίσει παραμέτρους διαμόρφωσης στο switch. Το switch ανταποκρίνεται μόνο σε ερωτήσεις από τον ελεγκτή.
- *Modify-State*: Στέλνονται από τον ελεγκτή για να χειριστούν την κατάσταση των switch. Κύριος σκοπός είναι να προσθέσουν, να διαγράψουν και να διαμορφώσουν καταχωρήσεις ροής ή ομάδας και να εισάγουν ή να διαγράψουν σύνολα εντολών μίας ομάδας στους πίνακες αλλά και να ορίσουν τις ιδιότητες των θυρών.
- *Read-State*: Χρησιμοποιούνται από τον ελεγκτή για την συλλογή διάφορων πληροφοριών όπως είναι, η τρέχουσα διαμόρφωση, τα στατιστικά και οι ιδιότητες του switch.
- *Packet-out*: Τα μηνύματα αυτά επιτρέπουν στον ελεγκτή να υποδείξει στο switch μέσω ποιας συγκεκριμένης θύρας να προωθήσει ένα πακέτο.
- *Barrier*: Ο ελεγκτής χρησιμοποιεί αυτά τα μηνύματα ώστε να διασφαλίσει ότι ισχύουν οι προϋποθέσεις για κάποιο συγκεκριμένο μήνυμα ή για να λάβει ειδοποιήσεις για την ολοκλήρωση μίας λειτουργίας.
- *Role-Request*: Χρησιμοποιούνται από τον ελεγκτή για να ορίσει τον ρόλο του καναλιού και την ταυτότητα του ελεγκτή ή και για να ζητήσει και τα δύο.
- *Asynchronous-Configuration*: Χρησιμοποιείται από τον ελεγκτή για να ορίσει ένα επιπλέον φίλτρο στα ασύγχρονα μηνύματα που επιθυμεί να λάβει, κάτι το οποίο είναι χρήσιμο όταν το switch συνδέεται σε πολλούς ελεγκτές.

2. **Asynchronous:** Στέλνονται από το switch χωρίς ο ελεγκτής να έχει ζητήσει κάτι τέτοιο. Συνήθως δηλώνουν την αλλαγή της κατάστασης ενός switch, την άφιξη ενός πακέτου κ.α. Παρακάτω είναι οι τέσσερις βασικές υποκατηγορίες μηνυμάτων.

- *Packet-in*: Κάθε νέο πακέτο που εισέρχεται στο switch και δεν αντιστοιχίζεται με καμία από τις υπάρχουσες εγγραφές ροής, προκαλεί την δημιουργία και αποστολή ενός μηνύματος Packet-in προς τον ελεγκτή (packet-in event). Αν το switch έχει αρκετή διαθέσιμη μνήμη ώστε να αποθηκεύσει προσωρινά (buffer) το πακέτο αυτό, τότε το μήνυμα που θα σταλεί θα περιλαμβάνει 128 bytes με τις απαραίτητες πληροφορίες που χρειάζεται ο ελεγκτής. Οι πληροφορίες αυτές αφορούν τις τιμές των κεφαλίδων του πακέτου που εισήλθε, καθώς και μία τιμή αναγνώρισης (buffer ID) του. Σε περίπτωση που το switch δεν υποστηρίζει την προσωρινή αποθήκευση πακέτων, ή δεν έχει αρκετή διαθέσιμη μνήμη, τότε το μήνυμα που θα αποσταλεί στον ελεγκτή θα περιλαμβάνει ολόκληρο το αρχικό πακέτο.
- *Flow-Removed*: Ενημερώνει τον ελεγκτή για την αφαίρεση μίας ροής από έναν πίνακα. Τα συγκεκριμένα μηνύματα δημιουργούνται σαν αποτέλεσμα της απαίτησης του ελεγκτή για διαγραφή μίας ροής. Επίσης υπαγορεύεται στο switch μετά από πόσο χρόνο αδράνειας μπορεί να διαγράψει αυτή την ροή.

- *Port-status*: Ενημερώνει τον ελεγκτή για μια αλλαγή σε μία θύρα. Το switch αναμένεται να στείλει μηνύματα σχετικά με την κατάσταση των θυρών στους ελεγκτές, για παράδειγμα όταν ένας χρήστης του switch απενεργοποιεί μία συγκεκριμένη θύρα ή όταν η σύνδεση μίας συγκεκριμένης θύρας πέσει.
- *Role-status*: Ενημερώνει τον ελεγκτή για την αλλαγή του ρόλου του. Όταν ένας νέος ελεγκτής επιλέγει τον εαυτό του σαν master, το switch στέλνει role-status μηνύματα στον προηγούμενο master ελεγκτή.
- *Controller-status*: Γνωστοποιεί στον ελεγκτή για το πότε η κατάσταση ενός OpenFlow καναλιού αλλάζει. Το switch στέλνει μηνύματα σε όλους τους ελεγκτές όταν η κατάσταση ενός καναλιού σε οποιοδήποτε switch αλλάζει. Αυτό μπορεί να βοηθήσει στην επεξεργασία ανακατεύθυνσης στην περίπτωση που οι ελεγκτές δεν έχουν την δυνατότητα να επικοινωνήσουν μεταξύ τους.
- *Flow-monitor*: Ενημερώνει τον ελεγκτή για μία αλλαγή σε ένα πίνακα ροής. Ο ελεγκτής μπορεί να ορίσει μία σειρά από οθόνες για να παρακολουθεί τις αλλαγές στους πίνακες.

Συμμετρικά

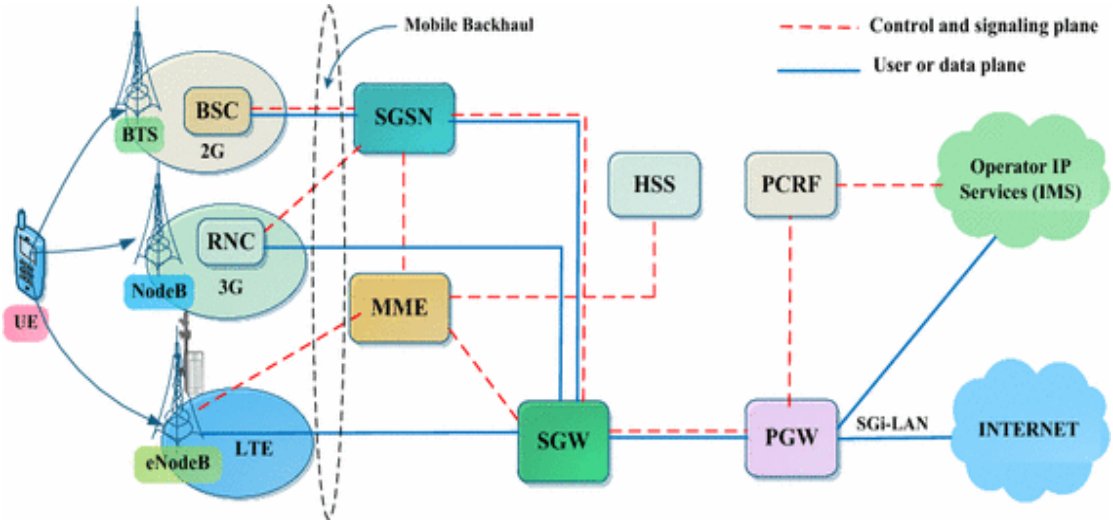
- *Hello*: Ανταλλάσσονται μεταξύ του switch και του ελεγκτή κατά την εγκατάσταση της σύνδεσης.
- *Echo*: Μηνύματα του τύπου echo request/reply μπορεί να αποστείλει οποιαδήποτε από τις δύο πλευρές και χρησιμοποιούνται για μετρήσεις καθυστέρησης (latency) και εύρους ζώνης (bandwidth). Ακόμη, χρησιμοποιείται για να επιβεβαιωθεί αν η μεταξύ τους σύνδεση είναι ενεργή.
- *Error*: Τα μηνύματα λάθους χρησιμοποιούνται είτε από τον ελεγκτή είτε από το switch για να γνωστοποιήσουν ένα πρόβλημα στην άλλη πλευρά της σύνδεσης. Συνήθως στέλνονται από το switch για να σηματοδοτήσουν μία αποτυχία σε αίτημα που ξεκίνησε από τον ελεγκτή.
- *Experimenter*: Ο σκοπός αυτών των μηνυμάτων είναι να παρέχουν περαιτέρω λειτουργικότητα, όσον αφορά τους τύπους των OpenFlow μηνυμάτων. Υλοποιήθηκαν κυρίως για στοιχεία μελλοντικών εκδόσεων του OpenFlow.

Κεφάλαιο 3

SDN ΣΤΑ ΚΙΝΗΤΑ ΚΑΙ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ

3.1 Σημερινά Κυψελωτά Δίκτυα (LTE)

Το LTE δίκτυο χωρίζεται σε δύο μέρη: το τμήμα του LTE ασχολείται με τις τεχνολογίες της ασύρματης πρόσβασης, ενώ ο εξελιγμένος πυρήνας πακέτων (Evolved Packet Core) ασχολείται με την τεχνολογία που σχετίζεται με το κυρίως δίκτυο. Ο χρήστης (UE) συνδέεται αρχικά σε ένα Σταθμό Βάσης (eNodeB) μέσω μίας ασύρματης διεπαφής. Ο ΣΒ πραγματοποιεί λειτουργίες διαχείρισης των ράδιο πόρων, όπως είναι για παράδειγμα η παροχή πόρων αλλά και η διαχείριση της παρεμβολής μεταξύ γειτονικών κυψελών. Ο ΣΒ συνδέεται σε μία πύλη εξυπηρέτησης (Serving Gateway) μέσω του δικτύου backhaul. Το S-GW αποτελεί σημείο σύνδεσης και εξυπηρέτησης για πολλούς ΣΒ. Το S-GW συνδέεται με μία πύλη δικτύου πακέτων (Packet Network Gateway), η οποία είναι υπεύθυνη για την απόδοση IP διευθύνσεων στους χρήστες, την εφαρμογή πολιτικών, το φιλτράρισμα των πακέτων αλλά και για την εφαρμογή χρέωσης. Στην ουσία είναι το σημείο τερματισμού για την μεταφορά των πακέτων από το LTE δίκτυο προς το εξωτερικό δίκτυο. Η οντότητα διαχείρισης της κινητικότητας (Mobility Management Entity) είναι υπεύθυνη για την διατήρηση της κινητικότητας των χρηστών αλλά και για την μεταφορά της κίνησης τους. Ο κεντρικός διακομιστής συνδρομητών (Home Subscriber Server) είναι μία κεντρική βάση δεδομένων όπου είναι αποθηκευμένα τα προφίλ των χρηστών, είναι υπεύθυνη για την πιστοποίηση της ταυτότητας του χρήστη. Η λειτουργία πολιτικής και κανόνων χρέωσης (Policy and Charging Rules Function) παρέχει κανόνες χρέωσης και ποιότητας υπηρεσίας στο P-GW. Τα πακέτα των χρηστών προωθούνται μέσω του ΣΒ και του P-GW κάνοντας χρήση του GTP (GPRS tunneling protocol) πρωτοκόλλου.



Εικόνα 10:Τοπολογία ενός LTE κυψελωτού δικτύου.

Εκτός από τις λειτουργίες στο επίπεδο των δεδομένων, οι Σταθμοί Βάσης, τα S-GWs και τα PGWs συμμετέχουν σε αρκετά πρωτόκολλα του επιπέδου ελέγχου. Σε συνεργασία με την Οντότητα Διαχείρισης της Κινητικότητας πραγματοποιούν σηματοδοσία hop-by-hop για διάφορες λειτουργίες όπως είναι, η εγκατάσταση και ο τερματισμός μίας συνόδου, η επαναδιαμόρφωση, η ενημέρωση σχετικά με την τοποθεσία και η μεταπομπή. Για παράδειγμα, σαν απάντηση στην απαίτηση ενός χρήστη για εγκατάσταση μίας συνόδου (π.χ VoIP κλήση) το P-GW στέλνει πληροφορίες QoS αλλά και πληροφορίες συνόδου στο S-GW, αυτό με την σειρά του προωθεί την πληροφορία στο MME, το οποίο θα ζητήσει από τον ΣΒ να παρέχει κάποιους συγκεκριμένους ράδιο πόρους και να ξεκινήσει την εγκατάσταση της σύνδεσης με το UE. Κατά την μεταπομπή ενός UE, ο ΣΒ που εξυπηρετεί την δεδομένη χρονική στιγμή τον χρήστη στέλνει ένα αίτημα μεταπομπής σε ένα άλλο ΣΒ. Αφού λάβει ένα acknowledgement μεταφέρει τον χρήστη στο ΣΒ προορισμού. Ο ΣΒ προορισμού ενημερώνει το MME ότι ο χρήστης έχει αλλάξει κυψέλη, έτσι ο αρχικός ΣΒ μπορεί να ελευθερώσει τους πόρους που είχε δεσμεύσει προηγουμένως για να εξυπηρετήσει τον χρήστη.

Το S-GW και το P-GW συμμετέχουν σε πρωτόκολλα δρομολόγησης όπως είναι το OSPF για παράδειγμα. Το PCRF διαχειρίζεται τις χρεώσεις σύμφωνα με τις ροές στο P-GW. Το PCRF παρέχει επίσης QoS εξουσιοδότηση σύμφωνα με την οποία παίρνεται η απόφαση για το πως θα αντιμετωπιστεί η κάθε ροή με βάση το προφίλ του κάθε συνδρομητή. Οι πολιτικές QoS μπορεί να είναι δυναμικές, για παράδειγμα με βάση την ημέρα και τον χρόνο. Ο HSS διατηρεί πληροφορίες για κάθε χρήστη, τέτοιες μπορεί να είναι το QoS προφίλ του κάθε συνδρομητή, οποιοιδήποτε περιορισμοί πρόσβασης που αφορούν την περιαγωγή κ.α. Σε περιόδους συμφόρησης ο ΣΒ μειώνει το μέγιστο επιτρεπόμενο ρυθμό για κάθε συνδρομητή σύμφωνα με το προφίλ του χρήστη και σε συνεργασία με το P-GW.

Η τωρινή αρχιτεκτονική των κυψελωτών δικτύων έχει σημαντικούς περιορισμούς. Οι λειτουργίες του επιπέδου των δεδομένων όπως είναι η παρακολούθηση, ο έλεγχος πρόσβασης και το QoS τοποθετούνται στο P-GW, δημιουργώντας ζητήματα που αφορούν την κλιμάκωση. Παράλληλα καθιστούν τον εξοπλισμό απαγορευτικά ακριβό (π.χ. 6 εκατομμύρια δολάρια για ένα Cisco packet gateway). Επίσης, η τοποθέτηση των παραπάνω λειτουργιών ανάμεσα στα όρια της κυψέλης και του υπόλοιπου διαδικτύου, αναγκάζει στην ουσία όλη την κίνηση να περάσει από το P-GW. Αυτό ισχύει και για την κίνηση μεταξύ χρηστών που βρίσκονται στην ίδια κυψέλη. Επιπλέον, η διαμόρφωση των διεπαφών του δικτυακού εξοπλισμού είναι συγκεκριμένη και εξαρτάται από το μοντέλο και τον κατασκευαστή που χρησιμοποιείται. Με άλλα λόγια υπάρχει εξάρτηση από τον κατασκευαστή, έτσι στην περίπτωση που οι πάροχοι θελήσουν να εισάγουν μία νέα υπηρεσία μπορεί να χρειαστεί να αντικαταστήσουν τον εξοπλισμό που διαθέτουν. Στην πραγματικότητα οι πάροχοι δεν έχουν άμεσο έλεγχο της λειτουργία του δικτύου τους.

3.2 Μελλοντική Αρχιτεκτονική Των Κυψελωτών Δικτύων

Η αρχιτεκτονική των μελλοντικών κυψελωτών δικτύων βασισμένη στο SDN θα έχει την δυνατότητα να χειρίζεται ολόκληρο το δίκτυο από μία κεντρική τοποθεσία και ταυτόχρονα θα υποστηρίζει πολλούς συνδρομητές, την συχνή κινητικότητα του χρήστη αλλά και να παρέχει λεπτομερή έλεγχο και μετρήσεις. Η SDN αρχιτεκτονική θα πρέπει να αντιμετωπίσει τις προκλήσεις προσαρμογής και κλιμάκωσης σε πραγματικό χρόνο στις οποίες τα σημερινά δίκτυα συνήθως αποτυγχάνουν.

3.3 Προτεινόμενες Αρχιτεκτονικές SDN Κυψελωτών Δικτύων

Στη συνέχεια γίνεται μία αναφορά σε προτεινόμενες αρχιτεκτονικές που έχουν αναπτυχθεί τα τελευταία χρόνια και αφορούν είτε το κομμάτι της πρόσβασης είτε του κυρίως δικτύου των δικτύων κινητής.

3.3.1 SoftCell

Το SoftCell **[11]** στοχεύει στην βελτίωση της κλιμάκωσης και της ευελιξίας στο κυρίως (core) κυψελωτό δίκτυο. Σε ένα κυψελωτό δίκτυο η κίνηση ξεκινά από το σημείο της πρόσβασης όπου παρατηρείται χαμηλό όγκος κίνησης (συνήθως της τάξης των Gbps) και φτάνει στα P-GWs όπου η κίνηση που παρατηρείται είναι της τάξης των Tbps. Το SoftCell εκμεταλλεύεται αυτή την ασυμμετρία στην ταχύτητα και δίνει την δυνατότητα της ταξινόμησης πακέτων στα switch πρόσβασης. Το αποτέλεσμα είναι τα S-GWs και τα P-GWs να αντικαθίστανται με OpenFlow switch τα οποία πραγματοποιούν λειτουργίες προώθησης μόνο. Επίσης, η λειτουργία του επιπέδου ελέγχου χωρίζεται μεταξύ του κεντρικού ελεγκτή και των τοπικών agents που υπάρχουν σε κάθε ΣΒ. Η λειτουργία της ταξινόμησης πακέτων πραγματοποιείται από τους agent, αυτό έχει σαν αποτέλεσμα την μείωση της κίνησης προς και από τον ελεγκτή από την στιγμή που τα πακέτα χειρίζονται από το επίπεδο των δεδομένων. Το αποτέλεσμα είναι η μείωση του κόστους που απαιτείται για εξοπλισμό. Τα κύρια μέρη του SoftCell είναι:

Α) *Ελεγκτής*: Εφαρμόζει πολιτικές υψηλού επιπέδου εγκαθιστώντας κανόνες στα switch τα οποία κατευθύνουν την κίνηση μέσω middleboxes. Οι πολιτικές κάθε φορά εφαρμόζονται σύμφωνα με τα χαρακτηριστικά του συνδρομητή και τον τύπο της εφαρμογής.

Β) *Access switch*: Κάθε ΣΒ έχει ένα switch πρόσβασης το οποίο ταξινομεί λεπτομερώς τα πακέτα κίνησης που έρχονται από τον χρήστη. Τέτοια switch μπορεί να είναι ένα software switch το οποίο τρέχει σε ένα Server. Ο Server με την σειρά του έχει ένα τοπικό agent (Local Agent) ο οποίος λαμβάνει ταξινομημένα πακέτα των χρηστών για λογαριασμό του κεντρικού ελεγκτή.

Γ) *Core switch*: Το υπόλοιπο δίκτυο αποτελείται από το switch κορμού συμπεριλαμβανομένου και κάποιων gateway switch που χρησιμοποιούνται για την δρομολόγηση της κίνησης στο Internet. Τα switch κορμού είναι OpenFlow switch και προωθούν την κίνηση μέσω κατάλληλων middlebloxes. Επίσης, τα gateway switch του SoftCell είναι φθηνότερα από τα τωρινά P-GWs, πραγματοποιούν απλή προώθηση των πακέτων και περιμένουν από τα middleboxes να επεξεργαστούν περαιτέρω τα πακέτα.

Δ) Middleboxes:Το SoftCell υποστηρίζει middleboxes του εμπορίου τα οποία μπορεί να είναι είτε αποκλειστικές συσκευές, είτε εικονικές μηχανές (virtual machines) ή και κανόνες επεξεργασίας των πακέτων στα switch. Κάθε λειτουργία (π.χ τείχος προστασίας) ενδέχεται να είναι διαθέσιμη σε διαφορετικές τοποθεσίες. Απαιτούν τα πακέτα να περνάνε από την συσκευή και στις δύο κατευθύνσεις.

3.3.2 SoftRan

Σε ένα πυκνό ασύρματο περιβάλλον με αρκετούς ασύρματους χρήστες και περιορισμένο φάσμα, είναι δύσκολη η παροχή ράδιο-πόρων, η πραγματοποίηση μεταπομπών, η διαχείριση της παρεμβολής καθώς και η εξισορρόπηση της κίνησης μεταξύ των κυψελών. Το SoftRan **[10]** ασχολείται με το δίκτυο πρόσβασης, η κύρια λειτουργία που έχει είναι η αφαίρεση των τοπικών ΣΒ σε μια γεωγραφική περιοχή και η μετατροπή τους σε ένα μεγάλο εικονικό ΣΒ ο οποίος αποτελείται από ένα κεντρικό ελεγκτή και ανεξάρτητους φυσικούς ΣΒ. Η εφαρμογή της παραπάνω αρχιτεκτονικής εμπεριέχει δύο προκλήσεις. Πρώτον, τον σχεδιασμό ενός ελεγκτή ο οποίος θα μπορεί να παρέχει μία δομή για την εκτέλεση διαφορετικών αλγορίθμων ελέγχου. Η καθυστέρηση μεταξύ του ελεγκτή και των ράδιο στοιχείων δεν θα έχει αρνητικό αντίκτυπο στην απόδοση.

3.3.3 CellSDN

Το CellSDN **[12]** παρουσιάζει μία SDN αρχιτεκτονική ειδικά σχεδιασμένη για κυψελωτά δίκτυα. Ο SDN ελεγκτής αποτελείται από ένα λειτουργικό σύστημα δικτύου (Network Operation System) το οποίο εκτελεί ένα σύνολο εφαρμογών όπως είναι η διαχείριση των ράδιο πόρων, η διαχείριση της κινητικότητας, η δρομολόγηση κ.α. Επιπλέον, για την βελτίωση της κλιμάκωσης στο επίπεδο ελέγχου αλλά και για να ανταποκριθεί στις ανάγκες των συχνών ενημερώσεων, κάθε switch έχει εγκατεστημένο ένα τοπικό agent ο οποίος πραγματοποιεί αποφάσεις σε πραγματικό χρόνο.

Αρχικά, οι εφαρμογές του ελεγκτή θα πρέπει να είναι σε θέση να εφαρμόζουν μία πολιτική με βάση τα χαρακτηριστικά των συνδρομητών, αντί της IP διεύθυνσης ή της φυσικής τοποθεσίας. Δεύτερον, για την βελτίωση της κλιμάκωσης κάθε switch εμπεριέχει έναν τοπικό agent ο οποίος πραγματοποιεί απλές δράσεις (παρακολούθηση της κίνησης και σύγκριση με ένα κατώφλι) κατ' εντολή του ελεγκτή.

Τρίτον, τα switch θα πρέπει να υποστηρίζουν πιο ευέλικτη ταξινόμηση πακέτων κάνοντας χρήση της επιθεώρησης πακέτων (deep packet inspection) αλλά και επιπρόσθετες δράσεις όπως είναι η συμπίεση της κεφαλίδας.

3.4 OpenFlow-SDN Στα Κυψελωτά Δίκτυα

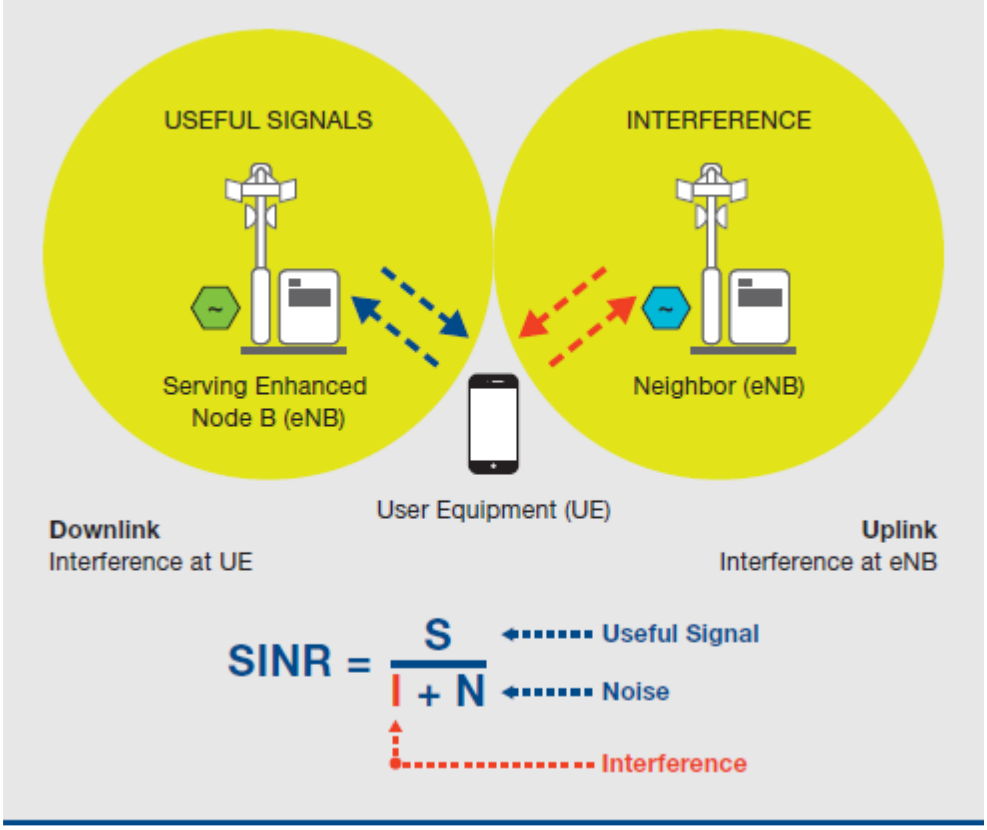
Ο ONF έχει ορίσει ένα αρχιτεκτονικό μοντέλο που απευθύνεται σε κυψελωτά δίκτυα **[5]**. Ο ONF περιγράφει δύο περιπτώσεις χρήσης όπου γίνεται εμφανές τα οφέλη που έχουμε από την χρήση του SDN με βάση το OpenFlow:

- Διαχείριση των παρεμβολών
- Διαχείριση της κίνησης

3.4.1 Διαχείριση Των Παρεμβολών Μεταξύ Των Κυψελών

Τα τελευταία χρόνια, υπάρχει τεράστια αύξηση της τηλεπικοινωνιακής κίνησης μαζί με την ραγδαία υιοθέτηση των έξυπνων κινητών συσκευών. Το LTE είναι το πιο πρόσφατο πρότυπο ασύρματης επικοινωνίας. Το LTE δίνει τη δυνατότητα για επικοινωνίες δεδομένων υψηλής ταχύτητας οι οποίες φτάνουν μέχρι και τα 150 Mbps, και υποστηρίζει την ολοένα και αυξανόμενη ανάγκη για ευρυζωνικές υπηρεσίες. Καθώς τα LTE δίκτυα αυξάνονται αντικαθιστώντας τα δίκτυα 3^{ης} γενιάς (3G), ταυτόχρονα αυξάνεται και η δικτυακή κίνηση. Ωστόσο, παρατηρείται ολοένα και περισσότερο το φαινόμενο της αύξησης των παρεμβολών μεταξύ των κεψελών. Η παρεμβολή οδηγεί σε σημαντική υποβάθμιση στην ποιότητα υπηρεσίας για τους χρήστες κινητές συσκευών, όπως φαίνεται στην παρακάτω εικόνα. Συνήθως, το αποτέλεσμα είναι η επικάλυψη των κυψελών μεταξύ γειτονικών σταθμών βάσης. Για την αποφυγή του φαινομένου αυτού, οι σταθμοί βάσης χρειάζεται να συντονίσουν την κατανομή των υποφερόντων (subcarriers). Ο στόχος, όπως απεικονίζεται στην εικόνα 11, είναι η μείωση το λόγο σήματος προς παρεμβολή συν τον θόρυβο (SINR) .Αυτό επιτυγχάνεται με χρήση τεχνικών διαχείρισης της παρεμβολής.

Στα LTE δίκτυα ο ΣΒ μπορεί να χρησιμοποιήσει όλα τα υποφέροντα. Ωστόσο, οι ΣΒ πρέπει να συντονίσουν την κατανομή των υποφερόντων τους για την αποφυγή παρεμβολών μεταξύ των χρηστών. Η διαχείριση της παρεμβολής στα LTE δίκτυα γίνεται με ένα κατανεμημένο τρόπο. Συγκεκριμένα, ο ΣΒ ειδοποιεί τους γειτονικούς σταθμούς σε περιπτώσεις που χρειάζεται υψηλότερη ισχύ για να μεταδώσει προς ένα χρήστη χρησιμοποιώντας ένα σύνολο των υποφερόντων. Αντίστοιχα, σε περιπτώσεις που λαμβάνει υψηλή παρεμβολή σε συγκεκριμένα υποφέροντα ειδοποιεί τους γειτονικούς ΣΒ να μειώσουν την ισχύ μετάδοσης τους.



Εικόνα 11: Η παρεμβολή στα κυψελωτά δίκτυα.

Υπάρχουν αρκετές τεχνικές σε χρήση στα LTE δίκτυα με σκοπό την αντιμετώπιση της παρεμβολής, κάποιες από αυτές είναι:

Inter-cell interference coordination (ICIC), η οποία επιλεκτικά μειώνει την ισχύ των υπό-καναλιών (subchannels) στο πεδίο της συχνότητας.

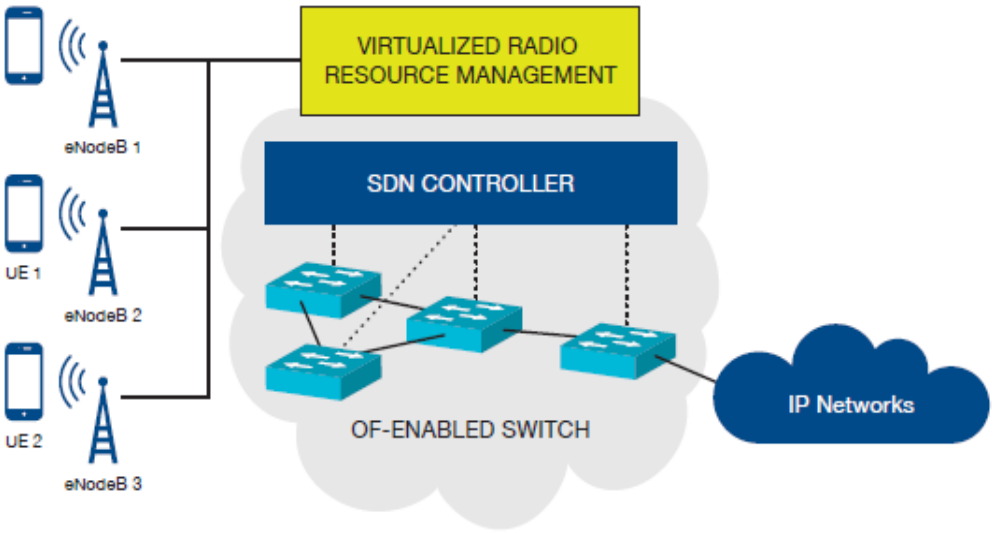
Enhanced inter-cell interference coordination (eICIC), όπου οι μακρό-κυψέλες συμπληρώνονται με πίκο-κυψέλες μέσα στην περιοχή κάλυψης τους (για hotspot σε δημόσιους χώρους όπως καφετέριες, αεροδρόμια, λιμάνια, τραίνα)

Coordinated multi-point transmission/reception (COMP), όπου η παρεμβολή μειώνεται για χρήστες στα όρια μίας κυψέλης με από κοινού σχεδιασμό αρκετών κυψελών με αρκετά ισχυρή παρεμβολή στα όρια της κυψέλης ή με από κοινού μετάδοση έτσι ώστε η λαμβανόμενη ισχύς και η δυνατότητα εξυπηρέτησης ενός χρήστη στα όρια μίας κυψέλης να μπορεί να βελτιωθεί.

Βέβαια υπάρχουν κάποια μειονεκτήματα στις τρέχουσες τεχνικές συντονισμού. Στα υπάρχοντα LTE δίκτυα όπως αναφέραμε και παραπάνω, η διαχείριση της παρεμβολής υλοποιείται με ένα κατανεμημένο τρόπο. Οι τεχνικές αντιμετώπισης, όπως είναι για παράδειγμα το COMP είναι βασισμένες σε λογισμικό, προσθέτοντας πολυπλοκότητα και απαιτώντας παραπάνω επεξεργασία. Αυτό έχει σαν αποτέλεσμα να χρειαζόμαστε περισσότερη ισχύ και απαιτήσεις σε πόρους στο δίκτυο της ράδιο πρόσβασης. Επιπλέον, οι αλγόριθμοι διαχείρισης της παρεμβολής μεταξύ των

κυψελών υλοποιούνται με χρήση κατανεμημένων αλγόριθμων χρωματισμού γράφων, λύση η οποία είναι σύνθετη.

Κάνοντας χρήση του SDN σε ένα LTE δίκτυο μπορούμε να ξεπεράσουμε διάφορους περιορισμούς σχετικά με την διαχείριση των παρεμβολών. Όπως φαίνεται στην εικόνα 12, το κεντρικό επίπεδο ελέγχου έχει μία συνολική άποψη για την κατάσταση του δικτύου. Έτσι, επιτρέπει οι αποφάσεις για την ανάθεση των ράδιο πόρων να παίρνονται μεταξύ πολλών σταθμών βάσης. Η συγκεκριμένη προσέγγιση είναι καλύτερη από την κατανεμημένη διαχείριση των πόρων (RRM), την διαχείριση της κινητικότητας, τα πρωτόκολλα δρομολόγησης και τις εφαρμογές που είναι σε χρήση σήμερα. Συγκεντρώνοντας την δικτυακή ευφυία, οι RRM αποφάσεις μπορούν να ρυθμιστούν σύμφωνα με την δυναμική ισχύ και το προφίλ της ανάθεσης των υπό-φερόντων του κάθε σταθμού βάσης. Επιπρόσθετα, η κλιμάκωση βελτιώνεται επειδή όσο καινούριοι χρήστες προστίθενται, η απαιτούμενη υπολογιστική ισχύ κάθε σταθμού βάσης παραμένει χαμηλά επειδή η RRM διαδικασία είναι κεντρική σε έναν SDN ελεγκτή.



Εικόνα 12: Κεντρικοποιημένος έλεγχος με χρήση τουOpenflow για διαχείριση παρεμβολών.

Θεωρώντας ότι ο ελεγκτής επικοινωνεί με τους σταθμούς βάσης μέσω του Open Flow, μπορούμε να έχουμε οποιεσδήποτε RRM αναβαθμίσεις ξεχωριστά για το υλικό κάθε σταθμού βάσης.

3.4.2 Διαχείριση της Κίνησης Των Κυψελών

Η καθοδήγηση της κίνησης (traffic steering) και η διαχείριση του μονοπατιού λαμβάνουν σημαντικής προσοχής από την SDN κοινότητα. Η καθοδήγηση της κίνησης είναι εφαρμόσιμη σε έναν αριθμό περιοχών και ενδεχόμενων περιπτώσεων χρήσης που περιλαμβάνουν την εξισορρόπηση φορτίου (Load balancing), την εφαρμογή φίλτρου σε περιεχόμενο (content filtering), τον έλεγχο πολιτικών, την αποφυγή και ανάκαμψη από καταστροφή (όπως η χρήση των μονοπατιών για backup).

Στο πλαίσιο των κινητών και ασύρματων δικτύων υπάρχουν περιπτώσεις χρήσης όπως είναι η αποσυμφόρηση (offloading), η περιαγωγή της Mobile κίνησης, η προσαρμογή στο περιεχόμενο (όπως adaptive streaming), καθώς και η βελτιστοποίηση της Mobile κίνησης που θα μπορούσαν να επωφεληθούν σημαντικά με την υλοποίηση του OpenFlow.

Όταν η φωνή κυριάρχησε, η σχεδίαση της κίνησης σε ένα RAN ήταν πιο προβλέψιμη. Η μετάβαση στα δεδομένα κινητής και η επακόλουθη αύξηση στην κίνηση ήχου και βίντεο είχε σαν αποτέλεσμα τρομακτικές απαιτήσεις σε εύρος ζώνης, οι οποίες αυξήθηκαν γρηγορότερα από τον προϋπολογισμό και τα μέσα έσοδα ανά χρήστη. Μία πιο ευέλικτη προσέγγιση χρειάζεται για την κλιμάκωση της χωρητικότητας ώστε να διασφαλιστεί βέλτιστη χρήση της RAN χωρητικότητας και την διάκριση των υπηρεσιών με σκοπό την άνοδο των εσόδων.

Για την περίπτωση αποσυμφόρησης της Mobile κίνησης, το SDN επιτρέπει την δυναμική τοποθέτηση ή επανατοποθέτηση της κυκλοφορίας στο δίκτυο σύμφωνα με κάποια κριτήρια. Τέτοια μπορεί να είναι ο αθροιστικός (ανά εφαρμογή, κυψέλη, εξοπλισμού χρήστη.κτλ.) ρυθμός ροής ή ο αθροιστικός αριθμός ροών σε μια συγκεκριμένη πόρτα/ζεύξη , η διάρκεια της ροής, ο αριθμός κινητών χρηστών ανά σταθμό βάσης, το διαθέσιμο εύρος ζώνης, η IP διεύθυνση, ο τύπος της εφαρμογής και η χρησιμοποίηση του εκάστοτε δικτυακού στοιχείου(NE).

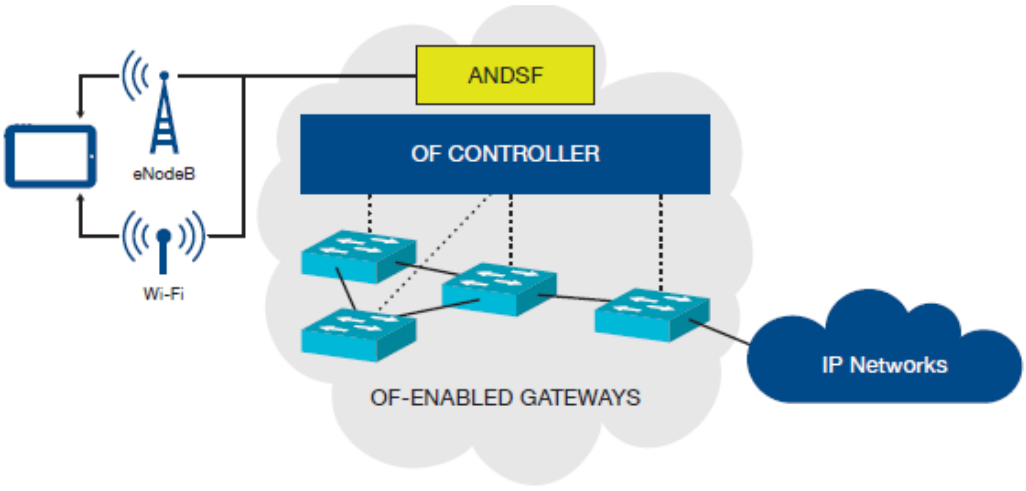
Υπάρχει η δυνατότητα τα κριτήρια να καθορίζονται είτε από το χρήστη, είτε από τον πάροχο της κινητής τηλεφωνίας. Για παράδειγμα, ο πάροχος της κινητής μπορεί να αποφασίσει ανάλογα με τις συνθήκες που επικρατούν εκείνη την στιγμή για την αποσυμφόρηση του δίκτυο του. Εναλλακτικά, οι χρήστες θα μπορούν να επιλέγουν με βάση τις προτιμήσεις τους. Για παράδειγμα, για τις φωνητικές κλήσεις δεν είναι εφικτή η διαδικασία της αποσυμφόρησης κάτι που μπορεί να γίνει όμως για την κίνηση δεδομένων. Επίσης, σε ένα προχωρημένο σενάριο, οι χρήστες θα μπορούν να συνδεθούν ταυτόχρονα σε πολλά δίκτυα (το πιθανότερο σε αυτά της επιλογής τους). Οι συνθήκες δικτύου στο πλαίσιο αυτό αφορούν την συμφόρηση του δικτύου (network congestion), την ποιότητα της υπηρεσίας (QoS) ή οποιοδήποτε άλλο τύπο ανατροφοδότησης ή μηχανισμό σχεδιασμού του δικτύου.

Οι παράγοντες ενεργοποίησης (όπως είναι το όριο στον ρυθμό ροής) μπορούν να οριστούν και να τροποποιηθούν δυναμικά, όσο η κίνηση παρακολουθείται από τον πάροχο της κινητής. Όταν για παράδειγμα, ο ρυθμός ροής υπερβεί τα 50 Kbps, θα γίνεται η μετατόπιση της ροής από το 4G στο Wi-Fi. Επίσης, διαφορετικά κριτήρια και

όρια μπορούν να εφαρμοστούν για διαφορετικές εφαρμογές που λειτουργούν στον ίδιο εξοπλισμό ενός χρήστη ή και για διαφορετικούς χρήστες. Να τονίσουμε ότι τα όρια θα μπορούν να βασιστούν σε ένα εύρος κριτηρίων όπως είναι για παράδειγμα, το προφίλ του χρήστη, η τοποθεσία και η υπηρεσία.

Με τον όρο αποσυμφόρηση εννοούμε την μεταφορά της κίνησης από ένα Mobile δίκτυο (cell, microcell, femtocell) σε ένα Wi-Fi δίκτυο το οποίο είναι γνωστό σαν Wi-Fi περιαγωγή. Η διαδικασία της μεταπομπής θα πρέπει να είναι εντελώς αδιαφανής στον χρήστη(καμία απώλεια δεδομένων, διατήρηση της IP διεύθυνσης), ώστε να επιτυγχάνεται διαρκής παροχή υπηρεσιών.

Η αποσυμφόρηση μπορεί να εφαρμοστεί και με αντίστροφο τρόπο (reverse offload). Κατά την διάρκεια συμφόρησης σε ένα Wi-Fi δίκτυο, αυτό μπορεί να μεταφέρει αυτόματα τους χρήστες σε ένα άλλο Wi-Fi δίκτυο ή σε κάποιο άλλο δίκτυο κινητής(3G ή 4G). Ο OpenFlow ελεγκτής, θα πρέπει να επικοινωνεί με κάποιες οντότητες όπως είναι η ανακάλυψη δικτύου πρόσβασης και επιλογή λειτουργίας (ANDSF,Access Network Discovery and Selection Function) για την ανακάλυψη ασύρματων δικτύων κοντά στον χρήστη καθώς και για την πραγματοποίηση της αποσυμφόρησης (Εικόνα 13). Η επιλογή του προορισμού περιαγωγής μπορεί να βασιστεί σε μια μέτρηση του QoS όπως είναι η απόδοση, η ισχύς του σήματος ή η απόσταση ώστε ο χρήστης να έχει ποιοτική υπηρεσία.



Εικόνα 13: Αποσυμφόρηση της κίνησης με χρήση του OpenFlow.

Για να εκμεταλλευτούμε την δυνατότητα που μας δίνει η αποσυμφόρηση απαιτείται η διασύνδεση του ελεγκτή με το ANDSF. Η διαδικασία της αποσυμφόρησης έχει γίνει ιδιαίτερα σημαντική με την αύξηση της κίνησης και των συσκευών επειδή επιτρέπει στους παρόχους να βελτιστοποιούν τους ράδιο πόρους αλλά και την ποιότητα εμπειρίας (QoE) για εφαρμογές που είναι απαιτητικές σε δεδομένα.

Μία ακόμη ενδιαφέρουσα εφαρμογή είναι η συνένωση των ασύρματων συνδέσεων, η οποία αποτελείται από την ομαδοποίηση των διαθέσιμων ασύρματων συνδέσεων ώστε να αυξηθεί η συνολική χωρητικότητα που είναι διαθέσιμη στον εξοπλισμό του χρήστη. Αυτό απαιτεί ο εξοπλισμός του χρήστη ή η κινητή συσκευή να είναι ικανή στον χειρισμό ταυτόχρονων αλλά και διαφορετικών ασύρματων συνδέσεων(Wi-Fi και 3G/4G).

3.5 Χρήση Του SDN Στα Ασύρματα Δίκτυα

3.5.1 Wireless SDN Πλεονεκτήματα

Αναμφισβήτητα το SDN έχει λάβει ιδιαίτερης υποστήριξης από τον ενσύρματο κόσμο και διάφορα παραδείγματα επιδεικνύουν την ευελιξία που επιτυγχάνει δημιουργώντας καινοτόμες εφαρμογές μερικές εκ των οποίων έχουν εφαρμογή και σε ασύρματα δίκτυα. Στη συνέχεια περιγράφονται μερικά από τα πλεονεκτήματα που μπορούμε να επιτύχουμε με την χρήση του SDN στα ασύρματα δίκτυα.

3.5.1.1 Βελτίωση Της Συνδεσιμότητας Του Τελικού Χρήστη Και QoS

Συχνά οι χρήστες των ασύρματων τεχνολογιών αντιμετωπίζουν απρόβλεπτες αλλαγές στην ποιότητα υπηρεσίας. Αυτό συμβαίνει συνήθως όταν ο χρήστης βρίσκεται στα όρια κάλυψης από ένα ΣΒ ή λόγω συμφόρησης του καναλιού επικοινωνίας που έχει επιλέξει. Στις περισσότερες περιπτώσεις υπάρχει ένα εναλλακτικό σημείο πρόσβασης κοντά στον χρήστη με λιγότερη συμφόρηση το οποίο ενδέχεται να διαχειρίζεται κάποιος άλλος.

Σε ένα SDN δίκτυο οι πολλαπλοί ελεγκτές για μία δεδομένη περιοχή κάλυψης, έχουν την δυνατότητα να επικοινωνούν μεταξύ τους ώστε να ανταλλάσσουν πληροφορίες που θα επιτρέπουν στους χρήστες να συνδεθούν σε οποιοδήποτε AP ανεξάρτητα από τον διαχειριστή στον οποίο ανήκει. Ένα AP το οποίο λαμβάνει ένα εισερχόμενο πακέτο με IP διεύθυνση πηγής που ανήκει σε ένα γειτονικό δίκτυο μπορεί να το προωθήσει στο οικείο δίκτυο χρησιμοποιώντας ένα μόνο κανόνα. Σε περίπτωση που χρειαστεί να αντιμετωπιστούν προβλήματα εξισορρόπησης φορτίου, ασφάλειας και ελέγχου πρόσβασης, η συνεργασία μεταξύ των διαχειριστών για μία συγκεκριμένη περιοχή θα εξασφαλίζει καλύτερο QoS για τον χρήστη.

Επιπλέον, για κάθε γεωγραφική περιοχή θα μπορούσε να οριστεί ένας ελεγκτής, συγκεντρώνοντας στατιστικά στοιχεία σχετικά με την χρησιμοποίηση των ασύρματων καναλιών όλων των APs ανεξαρτήτως του ποιος είναι ο διαχειριστής ή ο ιδιοκτήτης τους. Στη συνέχεια, ο ελεγκτής μπορεί να αποστείλει τα στατιστικά που αφορούν την συμφόρηση των καναλιών σε μία τερματική συσκευή. Ο χρήστης της τερματικής συσκευής με την σειρά του μπορεί να επιλέξει σε ποιο AP να συνδεθεί ή ποια ασύρματη διεπαφή να χρησιμοποιήσει για μία συγκεκριμένη εφαρμογή. Πράγματι, τα περισσότερα τερματικά κινητής τηλεφωνίας είναι εξοπλισμένα με διάφορες ασύρματες διεπαφές όπως Wi-Fi, 3G, 4G ή Bluetooth οι οποίες έχουν τα δικά τους χαρακτηριστικά σε όρους καθυστέρησης, σταθερότητας, απόδοσης και κάλυψης. Χωρίς ακριβή στατιστικά στοιχεία είναι δύσκολο για τα τερματικά να επιλέξουν

αυτόματα σε ποια τεχνολογία θα συνδεθούν και συνήθως χρησιμοποιούνται απλοί αλγόριθμοι για αυτή την επιλογή (π.χ επιλογή πάντοτε του Wi-Fi αντί μίας κυψέλης). Έχοντας τα κατάλληλα στατιστικά τα οποία αποστέλλονται από διαφορετικούς ελεγκτές ,το τερματικό είναι σε θέση να επιλέξει ποια τεχνολογία θα χρησιμοποιήσει για κάθε εφαρμογή **[4].**

3.5.1.2 Σχεδιασμός Πολλών Δικτύων

Το δίκτυο επωφελείται από την χρήση του SDN, ειδικότερα αν σκεφτούμε την περίπτωση λειτουργίας ενός ασύρματου καναλιού στο οποίο υπάρχει συμφόρηση. Για παράδειγμα, το Wi-Fi χρησιμοποιεί τρία μη αναδιπλούμενα κανάλια στην ζώνη των 2.4GHz. Έτσι, σε πυκνοκατοικημένες περιοχές είναι σύνηθες να υπάρχουν δεκάδες Wi-Fi δίκτυα που όλα παρεμβάλουν μεταξύ τους. Ο σχεδιασμός του δικτύου δεν είναι εφικτός διότι οι τοποθεσίες που βρίσκονται τα AP δεν ελέγχονται από κάποιον πάροχο.

Το SDN επιτρέπει την δημιουργία ελεγκτών με βάση την ζώνη που υπερβαίνουν τους διαχειριστές ενός δικτύου, υποδεικνύοντας τις επιλογές καναλιών και τον έλεγχο ισχύος στα AP με βάση τα επίπεδα αμοιβαίας παρεμβολής. Η κατανομή του καναλιού είναι ευρέως γνωστό πρόβλημα που ανάγεται σε χρωματισμό γράφων που σχηματίζονται από τη μοντελοποίηση των αλληλεξαρτήσεων των σημείων πρόσβασης.

Ο έλεγχος ισχύος συμβάλλει σημαντικά στην μείωση των παρεμβολών συνδέοντας τους χρήστες στο πλησιέστερο σημείο πρόσβασης. Παράλληλα, μειώνοντας την ισχύ μετάδοσης των δύο πλευρών, βελτιώνεται σημαντικά το πρόβλημα της παρεμβολής. Επίσης, η ικανότητα του SDN να συγκεντρώνει σε ένα μοναδικό εικονικό δίκτυο πολλούς ΣΒ διαφορετικών παρόχων συμβάλλει με την σειρά της στην επίλυση του προβλήματος.

Επίσης,τα AP συνήθως διαθέτουν περιορισμένο αριθμό επιπέδων μετάδοσης ισχύος. Αν θεωρήσουμε ένα σύνολο από APs σε μία περιοχή. Το SDN καθιστά δυνατό για κάθε AP την εύκολη αναγνώριση γειτονικών APs τα οποία είναι προσβάσιμα σε διαφορετικά επίπεδα μετάδοσης απλά στέλνοντας beacon frames. Όλα τα AP μπορούν να προωθήσουν τα συγκεκριμένα frame σε ένα ελεγκτή ο οποίος μπορεί να συσχετίσει αυτή την πληροφορία με την ισχύ μετάδοσης του πομπού (emitter). Αφού δοκιμαστούν όλες οι πιθανές τιμές μετάδοσης ισχύος, ο ελεγκτής μπορεί να δημιουργήσει όλα τα γραφήματα παρεμβολών και να επιλέξει το AP που μεγιστοποιεί την κάλυψη και ελαχιστοποιεί την παρεμβολή. Εάν για παράδειγμα διαθέτουμε n APs και κάθε ένα διαθέτει k επίπεδα ισχύος, ένας κύκλος επικοινωνίας απαιτεί n *k εκπομπές μηνυμάτων οι οποίες πραγματοποιούνται παράλληλα.

Εκτός από τις μεταπομπές, μπορούμε να πάρουμε σαν παράδειγμα τα δίκτυα ηλεκτρικής ενέργειας, όπου η αποσυμφόρηση είναι μία κοινή πρακτική αλλά απαιτεί πρώτα τον συντονισμό πριν την εκτέλεση της αποσυμφόρησης. Η ίδια διαδικασία μπορεί να ισχύσει και στα δίκτυα επικοινωνιών. Αρχικά, πρέπει να πραγματοποιηθεί η διαδικασία του συντονισμού και στη συνέχεια η διαδικασία της αποσυμφόρησης.

Σε περίπτωση που πραγματοποιηθεί η αποσυμφόρηση χωρίς την την λειτουργία του συντονισμού πρώτα, ενδέχεται να παρατηρηθεί το φαινόμενο της συμφόρησης. Επομένως, η παρουσία ενός ελεγκτή ο οποίος επιβλέπει την κατάσταση του δικτύου και αναλόγως ενεργοποιεί την αποσυμφόρηση καθιστά δυνατό την αποτροπή τέτοιων καταστάσεων.

3.5.1.3 Ασφάλεια

Στα ενσύρματα αλλά και στα ασύρματα δίκτυα, η δυνατότητα παρακολούθησης του SDN μπορεί να προσφέρει μία σαφή εικόνα σχετικά με την κατάσταση του δικτύου σε μία οντότητα που είναι υπεύθυνη για την ανίχνευση εισβολών ή για μη φυσιολογικές συμπεριφορές. Το φορτίο δικτύου και η κατανομή των πακέτων ανά πρωτόκολλο μπορούν να συγκριθούν με στατιστικά στοιχεία. Έτσι, μία διαδικασία θα μπορεί να αποφασίσει εάν η τρέχουσα κίνηση αντιστοιχεί στις αναμενόμενες τιμές για μία δεδομένη ημερομηνία και ώρα.

Στα ασύρματα δίκτυα η ανίχνευση επιθέσεων, όπως είναι για παράδειγμα η ύπαρξη ενός κακοπροαίρετου AP απαιτεί τη συνεργασία μεταξύ των APs, η οποία διευκολύνεται από την δυνατότητα παρακολούθησης και ελέγχου που προσφέρει το SDN. Ομοίως, σε mesh ή collaborative δίκτυα ο διαμοιρασμός πληροφοριών κατάστασης επιτρέπει την ανίχνευση κακόβουλων ή selfish χρηστών. Η δράση σε τέτοιες περιπτώσεις είναι η διαμόρφωση ενός απλού κανόνα στον ελεγκτή και η μεταφόρτωση του σε όλους τους διαχειριζόμενους κόμβους.

3.5.1.4 Εντοπισμός

Ο εντοπισμός των χρηστών έχει γίνει μία βασική λειτουργία για αρκετές υπηρεσίες που βασίζονται στην τοποθεσία. Η εμπειρία από τον εντοπισμό των smartphones έχει δείξει, ότι ένα τερματικό χρησιμοποιώντας το ασύρματο περιβάλλον του, μπορεί να επιτύχει αρκετά καλή ακρίβεια εντοπισμού, χρησιμοποιώντας μία βάση δεδομένων η οποία περιέχει τις θέσεις των APs, ακόμη και σε περιπτώσεις που δεν καθίσταται εφικτό με την χρήση του GPS. Ο ελεγκτής συλλέγει πληροφορίες από διάφορα AP και παρέχει στην υποδομή επαρκείς πληροφορίες για τον εντοπισμό του χρήστη, με τέτοια ακρίβεια για παράδειγμα ώστε να γίνει σωστή πρόβλεψη των μεταπομπών (handovers) αλλά και η παροχή υπηρεσίας με βάση την θέση.

3.6 SDN Στο 5G

Η κινητή όπως και η ασύρματη σύνδεση έχουν σημειώσει τεράστια αύξηση κατά την τελευταία δεκαετία. Έως τώρα, τα δίκτυα 3^{ης} και 4^{ης} γενιάς παρέχουν συνδεσιμότητα μέσω του κυρίως δικτύου IP (Evolved Packet Network). Επίσης, στοχεύουν στην παροχή απρόσκοπτης σύνδεσης μέσω κυψελωτών δικτύων όπως το 3G, το LTE, WLAN, Bluetooth. Παρόλα αυτά, η ολοένα και αυξανόμενη ζήτηση σε πόρους σε συνδυασμό με τα διαφορετικά πρότυπα κίνησης αλλά και την συνεχή αύξηση των κινητών συσκευών, είναι μερικοί από τους λόγους που έχουν αναγκάσει τους παρόχους αλλά και την επιστημονική κοινότητα στην υιοθέτηση των δικτύων 5^{ης}

γενίας. Συνεπώς, τα δίκτυα 5ης γενιάς θα πρέπει να παρέχουν μία βασική υποδομή με σκοπό την εξυπηρέτηση εκατοντάδων νέων συσκευών οι οποίες θα προστεθούν στο δίκτυο, ο τρόπος κίνησης των οποίων θα είναι εντελώς απρόβλεπτος. Οι πρωταρχικοί στόχοι των δικτυων 5G είναι, η αύξηση της χωρητικότητας, η αύξηση της ταχύτητας και η μείωση της καθυστέρησης. Τα δίκτυα 5G βρίσκονται προς το παρόν σε ερευνητικό στάδιο, με την επιστημονική κοινότητα να αναζητά αρχιτεκτονικές για την αντιμετώπιση των μελλοντικών προκλήσεων. Τα πλεονεκτήματα που προσφέρουν τα SDN δίκτυα αναμένεται να παίξουν κρίσιμο ρόλο στον σχεδιασμό και στην υλοποίηση των δικτύων 5G.

Τα δίκτυα 5G πρόκειται να σχεδιαστούν ώστε να είναι ανοιχτά, πιο ευέλικτα και ικανά να εξελίσσονται γρηγορότερα από τα παραδοσιακά δίκτυα. Επίσης, δεν θα βασίζονται σε τεχνολογίες δρομολόγησης και μεταγωγής. Ακόμη, θα είναι σε θέση να παρέχουν συγκλίνουσα επικοινωνία μεταξύ δικτύων διαφορετικών τεχνολογιών, δηλαδή την παροχή ενός ανοικτού συστήματος επικοινωνίας με σκοπό την επικοινωνία μεταξύ δορυφορικών συστημάτων, των κυψελωτών δικτύων, του cloud, των κέντρων δεδομένων, των οικιακών δικτύων καθώς και άλλων ανοικτών δικτύων και συσκευών. Επιπλέον, τα δίκτυα 5G θα είναι αυτόνομα και επαρκώς ικανά ώστε να προσαρμόζονται στις απαιτήσεις του εκάστοτε QoS, έτσι ώστε να μπορούν να χειριστούν δυναμικά δίκτυα που βασίζονται στις εφαρμογές (application-driven networks). Επιπρόσθετα η ασφάλεια, η ευελιξία όπως και η ακεραιότητα των δεδομένων θα είναι από τις βασικές προτεραιότητες στον σχεδιασμό των δικτύων 5G **[27]**.

Επιπλέον, τα δίκτυα 5G θα είναι σε θέση να χειρίζονται την κινητικότητα του χρήστη ώστε να του παρέχουν συνδεσιμότητα σε οποιαδήποτε κατάσταση. Παρόλα αυτά, το τερματικό του χρήστη θα είναι αυτό που θα αποφασίζει που θα συνδεθεί σε περιπτώσεις όπου υπάρχουν διαφορετικές τεχνολογίες πρόσβασης. Ακόμη, το τερματικό του χρήστη θα παραμένει ενεργό και θα αναζητά την τεχνολογία για σύνδεση με βάση τις δυναμικές αλλαγές που μπορεί να συμβαίνουν στην υπάρχουσα τεχνολογία πρόσβασης. Η υποδομή του ασύρματου δικτύου θα είναι βασισμένη στο SDN, το οποίο παρέχει την διευθέτηση της επικοινωνίας μεταξύ των εφαρμογών και των υπηρεσιών στο cloud και το τερματικό του χρήστη. Έτσι, το αποτέλεσμα θα είναι το δίκτυο να μπορεί να αντιμετωπιστεί δυναμικά με βάση τις πραγματικές ανάγκες και καταστάσεις που συμβαίνουν σε αυτό, ακόμη το δίκτυο θα έχει όφελος από την εικονικοποίηση των πόρων.

Με λίγα λόγια, ο στόχος των δικτύων 5G είναι η δημιουργία και η διατήρηση ενός δικτύου ικανού να χειρίζεται μεγαλύτερο αριθμό συσκευών, παρέχοντας συνδέσεις εκατό φορές πιο γρήγορα και καθυστέρηση μικρότερη ή ίση του 1 ms (millisecond). Οι υπάρχουσες αρχιτεκτονικές δεν προσφέρονται για την υλοποίηση ενός τέτοιου δικτύου, ο κύριος λόγος είναι τα διαφορετικά λειτουργικά συστήματα των κόμβων ενός δικτύου. Η βέλτιστη λύση θα είναι μία αρχιτεκτονική, η οποία θα προσφέρει εικονικοποίηση από άκρη σε άκρη αλλά και την δυνατότητα να διαχειριζόμαστε τα διαφορετικά μέρη ενός δικτύου σαν μία μονάδα. Εδώ είναι το σημείο όπου το SDN

μπαίνει στην εξίσωση. Τα δίκτυα SDN, συγκεντρώνουν την δικτυακή ευφυία σε ένα κεντρικό σημείο για την διαχείριση και την επιβολή πολιτικών για τα υπόλοιπα μέρη του δικτύου. Έτσι, τα δίκτυα 5G θα μπορούν να διαχειρίζονται από ένα κεντρικό σημείο, σε αντίθεση με τις τωρινές λύσεις όπου οι πάροχοι ή οι διαχειριστές ενός ασύρματου δικτύου πρέπει να διαμορφώσουν τον κάθε δικτυακό κόμβο ξεχωριστά.

3.6.1 Προτεινόμενες SDN Αρχιτεκτονικές Για Το 5G

Σε αυτή την ενότητα εξετάζονται αρχιτεκτονικές δικτύου αλλά και τεχνικές οι οποίες ενδέχεται να αξιοποιηθούν στα μελλοντικά δίκτυα 5G. Οι τεχνικές αυτές περιλαμβάνουν μη-ορθογώνια πολλαπλή πρόσβαση (NOMA), πολλαπλή είσοδο πολλαπλή έξοδο (MIMO), πλήρη αμφίδρομη λειτουργία (Full Duplex), επικοινωνία από συσκευή προς συσκευή (Device to Device), αυτοματοποιημένη οργάνωση δικτύου κ.α.

Στο **[28]** προτείνεται μία νέα αρχιτεκτονική με σκοπό την αξιοποίηση του SDN σε 5G δίκτυα, η οποία ονομάζεται SoftAir. Ο κύριος στόχος της συγκεκριμένης αρχιτεκτονικής είναι η αξιοποίηση των πλεονεκτημάτων που προκύπτουν από την χρήση του Cloud και της εικονικοποίησης στα δίκτυα 5G, με σκοπό την παροχή μίας κλιμακωτής, ευέλικτης και πιο ανθεκτικής αρχιτεκτονικής δικτύου.

Στην προτεινόμενη αρχιτεκτονική, το επίπεδο του ελέγχου διαχειρίζεται από δικτυακούς server και παρέχει εργαλεία διαχείρισης και βελτιστοποίησης για το επίπεδο των δεδομένων. Το επίπεδο των δεδομένων διαχειρίζεται από το κυψελωτό δίκτυο κορμού και αποτελείται από ΣΒ καθοριζόμενους από το λογισμικό (SD-BSs)

όπως και switches καθοριζόμενα από το λογισμικό (SD-switches) στο δίκτυο ασύρματης πρόσβασης. Ο ελεγκτής επιτελεί λειτουργίες Layer 1 ,2 και 3 σε υπολογιστές και σε απομακρυσμένα κέντρα δεδομένων.

Η συμβολή της SoftAir αρχιτεκτονικής μπορεί να κατηγοριοποιηθεί έχοντας τις εξής ιδιότητες. Πρώτον, δυνατότητα προγραμματισμού, οι SDN κόμβοι (SD-BSs, SD-switches) μπορούν να επαναπρογραμματιστούν ώστε να συνδέονται δυναμικά με διαφορετικούς πόρους του δικτύου. Δεύτερον, συνεργασία, οι κόμβοι μπορούν να υλοποιηθούν και να συνδεθούν σε κέντα δεδομένων ώστε να υπάρχει κοινός έλεγχος και βελτιστοποίηση με τελικό σκοπό την καλύτερη απόδοση του δικτύου. Τρίτον, εικονικοποίηση, έτσι ώστε πολλαπλά εικονικά ασύρματα δίκτυα μπορούν να υλοποιηθούν σε μία SoftAir πλατφόρμα, κάθε ένα από τα οποία λειτουργεί με βάση τα δικά του πρωτόκολλα και αλληλεπιδρά με τους διαθέσιμους πόρους του δικτύου χωρίς ταυτόχρονα να παρεμβαίνει σε κάποιο γειτονικό. Τέταρτον, ανοιχτό, τα δικτυακά στοιχεία του επιπέδου των δεδομένων (SD-BSs, SD-switches) έχουν κοινά πρωτόκολλα διεπαφής δεδομένων και ελέγχου, ανεξάρτητα από τις διαφορετικές τεχνολογίες προώθησης δεδομένων που παρέχονται από τους κατασκευαστές. Έτσι, η επίβλεψη και η διαχείριση του επιπέδου των δεδομένων μπορεί να απλοποιηθεί. Πέμπτον, ορατότητα (visibility), οι ελεγκτές είναι σε θέση να έχουν μία συνολική άποψη για το σύνολο του δικτύου συλλέγοντας πληροφορίες από το επίπεδο των δεδομένων των συσκευών.

Συνοψίζοντας, το SoftAir, προσπαθεί να προσφέρει μία ευέλικτη αρχιτεκτονική η οποία θα παρέχει μέγιστη φασματική αποδοτικότητα εκμεταλλευόμενη τα οφέλη που προκύπτουν από την εικονικοποίηση και το cloud, όπως επίσης και την σύγκλιση διαφορετικών δικτυακών στοιχείων από διαφορετικές εικονικές διεπαφές.

Στο **[29]** προτείνεται η χρήση ενός πολυ-επίπεδου (multi-tiered) cloud ελεγκτή με μηχανισμό επεξεργασίας συμβάντων για ασύρματα δίκτυα 5G. Για την παροχή του κατάλληλου ράδιο-επικοινωνιακού περιβάλλοντος που απαιτείται για την ασύρματη επικοινωνία στο 5G, οι τεχνικές του SDN σε συνδυασμό με το NFV μπορούν να βοηθήσουν ώστε να ξεπεραστεί η απομόνωση των ετερογενών δικτύων πρόσβασης όπως είναι το LTE, και το Wi-Fi. Επίσης, λόγω του ότι η φασματική απόδοση στο LTE βρίσκεται πολύ κοντά στα όρια χωρητικότητας του Shanonn, στην συγκεκριμένη αρχιτεκτονική αναφέρεται ότι ένας αποτελεσματικός τρόπος για την μείωση της κίνησης στα δίκτυα 5G, είναι η βελτίωση της πρόσβασης στα ετερογενή δίκτυα.

Μία σημαντική πρόκληση για τα ασύρματα δίκτυα 5G είναι η ποιότητα της εμπειρίας του χρήστη (QoE). Τα δίκτυα 5G θα χρειαστεί να εξυπηρετήσουν ένα μεγάλο αριθμό συσκευών με διαφορετικά πρωτόκολλα και απαιτήσεις σε QoS η κάθε μία. Επίσης, αρκετές μελλοντικές εφαρμογές, θα έχουν ως απαίτηση η καθυστέρηση (delay) να είναι της τάξης μερικών χιλιοστών του δευτερολέπτου. Με αυτόν τον τρόπο, οι μεταβαλλόμενες απαιτήσεις της εκάστοτε εφαρμογής για απόδοση (throughput), για καθυστέρηση αλλά και για τις διακυμάνσεις της καθυστέρησης (jitter), αυξάνουν την πολυπλοκότητα με σκοπό την παροχή πόρων για την οποιαδήποτε εφαρμογή.

Η κύρια συμβολή της συγκεκριμένης αρχιτεκτονικής είναι η παροχή ενός ασύρματου δικτύου πρόσβασης για το 5G το οποίο συνυπάρχει παράλληλα με άλλα ετερογενή ασύρματα δίκτυα. Επίσης, με την παρακολούθηση της κατάστασης του δικτύου σε πραγματικό χρόνο, σε περιπτώσεις που το δίκτυο αντιμετωπίζει προβλήματα συμφόρησης, πραγματοποιείται η αποστολή της κατάλληλης εντολής με βάσης τις απαιτήσεις του QoS.

Ένα ακόμη σημαντικό μέρος της συγκεκριμένης αρχιτεκτονικής είναι ο σχεδιασμός ενός cloud επιπέδου για το δίκτυο καθώς και η εφαρμογή δύο ελεγκτών: ένας ελεγκτής ο οποίος βρίσκεται στην άκρη του δικτύου (Edge Controller, EC) και ένας παγκόσμιος ελεγκτής (Global Controller, GC). Η λογική πίσω από αυτή την σχεδίαση είναι να μειωθεί ο χρόνος απόκρισης αλλά και να βελτιωθεί η εξισορρόπηση του όγκου εργασίας για τον cloud ελεγκτή. Ο EC διαχειρίζεται συμβάντα που αφορούν ένα συγκεκριμένο τομέα ενός ασύρματου δικτύου πρόσβασης, από την άλλη πλευρά ο GC διαχειρίζεται συμβάντα από διαφορετικά ασύρματα δίκτυα πρόσβασης.

Στο **[30]** εξετάζεται το γεγονός ότι το Διαδίκτυο χρησιμοποιεί ως επί το πλείστον μόνο μία διαδρομή μεταξύ των δύο τελικών σημείων επικοινωνίας, καθώς επίσης και ότι τα δίκτυα μεταγωγής πακέτων είναι η βασική υποδομή επικοινωνίας. Είναι προφανές, πως η χρήση μίας μόνο διαδρομής δεν είναι αρκετή ώστε να παρέχει ασφαλή επικοινωνία διασφαλίζοντας παράλληλα χαμηλό ποσοστό απωλειών, ο κύριος λόγος είναι πως η επικοινωνία είναι ευάλωτη σε περιπτώσεις που έχουμε

παραποίηση των πακέτων από κάποιον κακόβουλο χρήστη. Επίσης, σε περιπτώσεις όπου υπάρχουν μεγάλες αποστάσεις, η κατάσταση χειροτερεύει καθώς αυξάνεται η καθυστέρηση.

Στην συγκεκριμένη έρευνα γίνεται χρήση κωδικοποιημένων δικτύων (code centric networks), ώστε να επιτευχθεί μεγαλύτερη απόδοση, ασφάλεια αλλά και να μειωθεί ο χρόνος καθυστέρησης. Επιπλέον, επιτρέπεται στους routers να χρησιμοποιούν κωδικοποίηση δικτύου ανάλογα με την κατάσταση που επικρατεί εκείνη την στιγμή στο δίκτυο. Η προτεινόμενη αρχιτεκτονική ενδεχομένως να αποτελέσει ένα χρήσιμο εργαλείο για δίκτυα που παρουσιάζουν αρκετές απώλειες. Η λειτουργία κωδικοποίησης των SDN κόμβων ενδέχεται να βελτιώσει αρκετά την απόδοση του δικτύου.

Όπως φαίνεται, αρκετές έρευνες προσπαθούν να δώσουν μία άμεση και αξιόπιστη λύση σχετικά για τα δίκτυα 5G. Το SDN έχει διάφορους περιορισμούς, όπως και πλεονεκτήματα, για παράδειγμα ο διαμοιρασμός των πόρων και η διαχείριση μίας συνόδου (session management). Ο κύριος περιορισμός αφορά τις υπολογιστικές ικανότητες και τους πόρους των κινητών συσκευών. Ως αποτέλεσμα, επειδή οι χρήστες κινητών τηλεφώνων στέλνουν αιτήματα ξανά και ξανά προς τον ενσωματωμένο ελεγκτή για κανόνες ροής, το overhead αυξάνει σημαντικά. Το συμπέρασμα είναι ότι θα χρειαστούν επιπλέον έρευνες με σκοπό την αξιοποίηση του SDN για τα δίκτυα 5G.

Κεφάλαιο 4

Data Plane Timestamp

Σε αυτό το κεφάλαιο γίνεται ανάλυση του Data Plane Timestamp, μία τεχνική σημαντική για την αντιμετώπιση των προκλήσεων που προκύπτουν λόγω της φύσης της SDN αρχιτεκτονικής. Όπως αναφέραμε και στα προηγούμενα κεφάλαια, τα SDN δίκτυα ορίζουν μία κεντρικοποιημένη αρχιτεκτονική, όπου το επίπεδο ελέγχου ελέγχεται από έναν ελεγκτή. Η συγκεκριμένη προσέγγιση εμπεριέχει προκλήσεις που αφορούν την συνέπεια και την απόδοση ενός δικτύου. Σε ένα παραδοσιακό δίκτυο, ο διαχειριστής είναι υπεύθυνος για την υλοποίηση του δικτύου, για την επίβλεψη και την διαμόρφωση των συσκευών αλλά και για την επιβολή αλλαγών όταν αυτό κρίνεται απαραίτητο. Από την άλλη πλευρά, σε ένα SDN δίκτυο, ο ελεγκτής είναι υπεύθυνος για την εκτέλεση συχνών ενημερώσεων με σκοπό την διαμόρφωση του δικτύου. Επίσης, Ο ελεγκτής αναλαμβάνει να ελαχιστοποιήσει τις ανωμαλίες που μπορεί να προκύψουν στο δίκτυο κατά την διάρκεια μιας διαδικασίας ενημέρωσης. Ακόμη, οι ενημερώσεις πρέπει να σχεδιάζονται με γνώμονα την απόδοση και το μέγεθος ενός δικτύου, κάτι που συνεπάγεται ότι δεν πρέπει να είναι αρκετά περίπλοκες.

Ο χρόνος και ο συγχρονισμός των ρολογιών χρησιμοποιούνται από διάφορα δικτυακά πρωτόκολλα για διαφορετικούς λόγους, όπως είναι για παράδειγμα οι μετρήσεις που γίνονται σε ένα δίκτυο **[21]**. Για παράδειγμα, η μέτρηση της καθυστέρησης (latency measuring) καθώς και ο αριθμός των απολεσθέντων πακέτων (packet loss) σε ένα δίκτυο. Επίσης, μπορούν να χρησιμοποιηθούν για δικτυακές ενημερώσεις βασισμένες στον χρόνο **[14]** που μπορεί να αφορούν αλλαγή στην πολιτική ή την αλλαγή στην διαδρομή της κίνησης, αλλά και για την εξισορρόπηση του φορτίου.

Στο **[15]** προτείνεται η χρήση του Data Plane Timestamp, το πεδίο εφαρμογής της συγκεκριμένης τεχνικής είναι στο πεδίο των δεδομένων και αφορά την επικοινωνία των switch με τον αντίστοιχο ελεγκτή αλλά και των switch μεταξύ τους. Υποστηρίζεται ότι η προσθήκη ενός επιπρόσθετου χρονικού πλαισίου σε κάθε πακέτο μπορεί να το καταστήσει ένα χρήσιμο εργαλείο για διάφορες δικτυακές εφαρμογές. Η τεχνική είναι εμπνευσμένη από την επέκταση που υποστηρίζει η TCP κεφαλίδα **[20]** η οποία προσθέτει ένα timestamp πεδίο σε κάθε πακέτο όταν χρειάζεται να υπολογιστεί ο χρόνος άφιξης και επιστροφής ενός πακέτου.

Η συγκεκριμένη τεχνική εισάγει μία timestamp κεφαλίδα (*Data Plane Timestamp Header*) στην κεφαλίδα ενός υπάρχοντος πακέτου κάνοντας το ένα χρήσιμο εργαλείο για διαφορετικές εφαρμογές όπως είναι: η επίβλεψη της κατάστασης ενός δικτύου, η μέτρηση της καθυστέρησης, οι συνεχείς ενημερώσεις που γίνονται σε ένα δίκτυο, η εξισορρόπηση φορτίου αλλά και οι απώλειες πακέτων κάτι που με την χρησιμοποίηση του timestamp πεδίου δεν απαιτεί την ανταλλαγή επιπρόσθετων μεταδεδομένων μεταξύ των switch. Επίσης, έχει την δυνατότητα να παίξει τον ρόλο

της ετικέτας διαμόρφωσης μειώνοντας έτσι τον όγκο διαχείρισης για τον ελεγκτή. Τέλος επιτρέπει την χρήση του time-division με σκοπό την προώθηση μεγάλων ροών μέσω διαφορετικών διαδρομών προσφέροντας μεγαλύτερο throughput σε σύγκριση με άλλες μεθόδους εξισορρόπησης. Έτσι με την χρήση της DPT κεφαλίδας μπορούμε να πούμε πότε ένα πακέτο μπήκε στο δίκτυο. Επιπλέον, με την μέθοδο ετικέτας που χρησιμοποιείται από το DPT επιτρέπεται σε διαφορετικές εφαρμογές να το χρησιμοποιούν ταυτόχρονα. Έτσι τα switch έχουν την δυνατότητα να επεξεργάζονται κάθε πακέτο σύμφωνα με την τιμή που έχει το timestamp.

Η DPT τεχνική είναι συμβατή με ένα SDN δίκτυο για δύο λόγους. Το SDN είναι ένα τοπικό διαχειριζόμενο περιβάλλον, έτσι η DPT κεφαλίδα μπορεί να εισαχθεί από ένα switch εισόδου και να αφαιρεθεί από ένα switch εξόδου. Επιπλέον, η τάση που επικρατεί στα SDN δίκτυα είναι η προώθηση πακέτων ανεξάρτητα του πρωτοκόλλου. Προσφέροντας την ευελιξία για πρόσθεση και αφαίρεση οποιασδήποτε κεφαλίδας είναι δυνατή η λήψη αποφάσεων χρησιμοποιώντας οποιοδήποτε πεδίο κεφαλίδας. Έπομένως, τα switch μπορούν να πάρουν αποφάσεις που αφορούν την επεξεργασία του πακέτου λαμβάνοντας υπόψιν το DPT πεδίο.

Η διαδικασία ξεκινάει με το switch εισόδου το οποίο προσθέτει μία DPT κεφαλίδα σε κάθε πακέτο που εισέρχεται σε ένα SDN-based δίκτυο (ουσιαστικά το DPT πεδίο δημιουργείται από το switch εισόδου), τα switch προωθούν το πακέτο και χρησιμοποιούν το DPT πεδίο για να πάρουν αποφάσεις αντιστοίχισης μέχρι την στιγμή που η DPT κεφαλίδα αφαιρείται από το switch εξόδου. Από την στιγμή που οι διαδικασίες αντιστοίχισης στα SDN switch επιτρέπουν την επί μέρους κάλυψη των πεδίων κεφαλίδας (partially masked) **[3]** μπορούμε να ορίσουμε ένα χρονικό εύρος, περιορίζοντας με τον τρόπο αυτό πολιτικές ή διαδρομές σε ένα συγκεκριμένο χρονικό όριο.

Η DPT κεφαλίδα είναι 64-bit μορφής όμοια με την μορφή που έχει το timestamp στο Network Time Protocol. Αυτή η μορφή της ώρας αντιπροσωπεύει τον χρόνο που έχει περάσει από την ημερομηνία βάσης, για παράδειγμα 6 Δεκεμβρίου 2008. Αυτή η μορφή του timestamp πεδίου αποτελείται από δύο 32-bit πεδία, το ένα πεδίο αντιπροσωπεύει το ακέραιο μέρος των δευτερολέπτων και το δεύτερο το παραγοντικό μέρος.Κάθε bit μπορεί να λάβει τρεις τιμές: 0/ 1/ * (**=αντιπρωσοπεύει bits τα οποία είναι partially masked*).

Όσον αναφορά στο ερώτημα που έχει να κάνει με το κατά πόσο είναι πρακτική η προσθήκη μίας επιπλέον κεφαλίδας σε κάθε πακέτο η απάντηση είναι πως το SDN εφαρμόζεται σε δίκτυα που χρησιμοποιούν πρωτόκολλα επικάλυψης τα οποία παρέχουν επεκτασιμότητα για την προσθήκη πεδίων metadata κάτι που σημαίνει πως η χρήση του DPT είναι συμβατή .Επίσης σε αρκετές εφαρμογές η χρησιμοποίηση μόνο ενός bit είναι αρκετή. Το DPT παρέχει ένα πεδίο το οποίο μπορεί να χρησιμοποιηθεί για διαφορετικούς λόγους. Παρόλο που η προσθήκη μιας DPT κεφαλίδας σε κάθε πακέτο μπορεί να φαίνεται ότι προσθέτει επιπλέον κόστος, σε αρκετές περιπτώσεις το κόστος αυτό περιορίζεται στο 1 bit, συνεπώς τα οφέλη που αποκομίζονται από το DPT υπερτερούν του κόστους.

Κεφάλαιο 5

Προσομοίωση ενός SDN δικτύου χρησιμοποιώντας το Mininet και τον POX ελεγκτή

Σε αυτό το κεφάλαιο πραγματοποιήθηκε η προσπάθεια προσομοίωσης ενός SDN δικτύου και συγκεκριμένα του OpenFlow πρωτοκόλλου, κάνοντας χρήση του Mininet εξομοιωτή και παρουσιάζοντας τρεις διαφορετικές περιπτώσεις χρήσης του ελεγκτή του δικτύου (συγκεκριμένα ενός POX ελεγκτή).

Υπάρχουν αρκετοί λόγοι που κάνουν απαραίτητη την χρήση ενός εξομοιωτή όπως το Mininet **[16]**. Πρώτον, υπάρχουν λίγες δικτυακές συσκευές διαθέσιμες για την εφαρμογή του OpenFlow προτύπου καθώς δεν είναι μία διαδεδομένη τεχνολογία ακόμη. Επίσης, η εφαρμογή ενός τέτοιου δικτύου σε ένα μεγάλο αριθμό συσκευών είναι δύσκολη προς το παρόν. Έτσι, για να ξεπεράσουμε αυτούς τους περιορισμούς χρησιμοποιούμε κάποιες τεχνολογίες εξομοίωσης με το Mininet να είναι μία από τις πιο σημαντικές.

Το Mininet είναι ένας εξομοιωτής δικτύων, ο οποίος δημιουργεί δίκτυα εικονικών hosts, switch, ελεγκτών και συνδέσεων. Είναι μια απλή και οικονομική πλατφόρμα δοκιμών για την ανάπτυξη OpenFlow εφαρμογών και παράλληλα επιτρέπει σε διαφορετικούς προγραμματιστές να εργάζονται ανεξάρτητα, πάνω στην ίδια τοπολογία. Το Mininet παρέχει ένα σύνολο έτοιμων τοπολογιών, αλλά και την δυνατότητα παραμετροποίησης τους. Ο κώδικας που αναπτύσσεται και δοκιμάζεται στο Mininet έχει την δυνατότητα να μετακινείται απευθείας σε πραγματικά συστήματα, με λίγες μόνο μετατροπές. Συγκρίνοντας το με άλλους εξομοιωτές δικτύων, πλατφόρμες δοκιμών και προσομοιωτές, το Mininet υπερτερεί στον χρόνο εκκίνησης, στην κλιμάκωση, στο υψηλότερο εύρος ζώνης, αλλά και στο ότι είναι ευκολότερο να το εγκαταστήσει κάποιος.

Ο POX **[17]** ελεγκτής είναι προεγκατεστημένος στο Mininet και παρέχει ένα τρόπο επικοινωνίας με τα SDN switch, κάνοντας χρήση του OpenFlow πρωτοκόλλου. Είναι μία πλατφόρμα λογισμικού που έχει αναπτυχθεί σε Python και αποτελεί ένα δημοφιλές εργαλείο για διδασκαλία και έρευνα, στο πεδίο του SDN. Αποτελείται από επιμέρους συστατικά, τα οποία αποτελούν προγράμματα σε Python. Επίσης, μπορεί να χρησιμοποιηθεί ως ένας βασικός ελεγκτής ή σαν βάση για την δημιουργία ενός πολυπλοκότερου SDN ελεγκτή. Ο POX, χρησιμοποιεί την έκδοση 2.7 της Python και μπορεί να λειτουργήσει σε οποιαδήποτε λειτουργικό σύστημα Linux,Mac, Windows.

Για την προσομοίωση χρησιμοποιήθηκε:

- Ένας υπολογιστής Lenovo με επεξεργαστή Intel(R) Core(TM) i3-3217U CPU @1.80GHZ, 4GB RAM σε Windows 10 64 bit.
- Το VirtualBox VM version 5.1.22 της Oracle.

- Ο προσομοιωτής Mininet έκδοσης 2.2.2 σε λειτουργικό Linux Ubuntu 14.043 32 bit με 1GB RAM και ο POX ελεγκτής εγκαταστάθηκαν στον υπολογιστή μέσω του VirtualBox.
- Ένας ssh client και πιο συγκεκριμένα το Putty .
- Το λογισμικό Xming για προώθηση.

Για την χρήση του Mininet, αρχικά, έγινε η εγκατάσταση του VirtulaBox. Στη συνέχεια κατέβηκε το Mininet VM image και έγινε η δημιουργία μίας νέας εικονικής μηχανής στο VirtualBox. Αφού έγινε η εγκατάσταση του Mininet, έγινε προσθήκη από τις ρυθμίσεις ενός επιπρόσθετου host-only network adapter. Η εκκίνηση στο Mininet πραγματοποιείται επιλέγοντας VirtualBox Mininet και σαν username και password χρησιμοποιούμε το mininet. Επίσης χρειάζεται να κάνουμε εκκίνηση και το Xming. Αφού έχουμε κάνει είσοδο στο Mininet, δίνοντας την εντολή *ifconfig –a* παρατηρούμε ότι το eth0 είναι: 192.168.156.101, η οποία είναι η διεύθυνση σύνδεσης του VM με το host-only network.Ακόμη, απαιτείται η εκτέλεση του putty.exe, δίνοντας σαν παραμέτρους την IP:192.168.156.101 και port:22, επίσης απαιτείται και η ενεργοποίηση της επιλογής X11 forwarding.Στο παράθυρο που μας ανοίγει δίνουμε σαν username και password την εντολή mininet.

Κατά την διαδικασία της προσομοίωσης, μελετήθηκαν τρεις διαφορετικές περιπτώσεις χρήσης. Όπως έχουμε αναφέρει και στα προηγούμενα κεφάλαια, οι δικτυακές συσκευές ενός SDN δικτύου λειτουργούν μόνο σαν συσκευές προώθησης. Στην πρώτη περίπτωση, τρέχουμε το *hub.py* script στον POX ελεγκτή. Ο ελεγκτής μεταδίδει την συγκεκριμένη πληροφορία μέσω του OpenFlow στο switch, το οποίο γνωρίζει πλέον ότι όταν λάβει ένα πακέτο πρέπει να το προωθήσει έξω από όλες τις θύρες τους, εκτός από την θύρα από την οποία έχει λάβει το πακέτο. Στη συνέχεια δοκιμάζοντας ένα ping από τον host 1 προς τον host 2 παρατηρούμε ότι το switch προωθεί το πακέτο (εκτός από τον host 1) και στους host 2 και 3, εξαιτίας του ότι το switch λειτουργεί σαν hub.

Στην 2η περίπτωση, δίνουμε την εντολή στον ελεγκτή να τρέξει ένα script που ονομάζεται *l2_learning.py*. Η διαφορά με την 1η περίπτωση, είναι ότι ο ελεγκτής έχει διαμορφώσει το switch να λειτουργεί σαν συσκευή επιπέδου 2 (Layer 2), σε αντίθεση με την 1η περίπτωση όπου το switch λειτουργεί σαν συσκευή επιπέδου 1 (layer 1). Αρχικά, όταν πραγματοποιούμε ping από τον host 1 στον host 2, το switch δεν γνωρίζει πως να διαχειριστεί το πακέτο και το αποστέλλει μέσω του OpenFlow στον ελεγκτή, αυτός με την σειρά του διαμορφώνει τους κανόνες προώθησης στο switch. Έτσι, όταν επιχειρούμε ξανά ping από τον host 1 στον host 2, παρατηρούμε ότι ο host 2 είναι ο μοναδικός παραλήπτης του πακέτου.

Στην 3η περίπτωση γίνεται χρήση ενός firewall.py script σε Python. Το συγκεκριμένο script φιλτράρει την κίνηση των πακέτων με βάση την φυσική διεύθυνση (MAC address) του αποστολέα και του παραλήπτη. Πιο συγκεκριμένα, επιτρέπει στους hosts με MAC διευθύνσεις 00:00:00:00:00:1,και 00:00:00:00:00:2 να επικοινωνούν μεταξύ τους και οποιαδήποτε άλλο πακέτο με διαφορετική MAC διεύθυνση να απορρίπτεται.

5.1 Λειτουργία Ενός OpenFlow Switch Σαν Hub

Το hub είναι μια συσκευή επιπέδου 1, χρησιμοποιεί επεξεργασία φυσικού επιπέδου για την επεξεργασία και την προώθηση ενός πακέτου. Από την στιγμή που θα λάβει ένα πακέτο το προωθεί έξω από όλες τις θύρες που έχει. Για την δημιουργία της τοπολογίας.

Χρησιμοποιώντας την εντολή:

$ sudo mn –topo single,4 –mac –switch ovsk –controller remote

Δημιουργούμε μία τοπολογία με ένα switch, τέσσερις host συνδεδεμένους με το switch και ένα ελεγκτή. Δίνοντας την εντολή *pingall,* παρατηρούμε ότι δεν υπάρχει απάντηση από κάποιον host. Ο λόγος είναι ότι δεν έχουμε τρέξει κάποιο script στον ελεγκτή, ώστε αυτός με την σειρά του να ενημερώσει το switch για το πως θα πρέπει να χειριστεί ένα πακέτο.

Στη συνέχεια θα προγραμματίσουμε τον POX να κάνει χρήση ενός hub.py script, έτσι από την κονσόλα του Mininet δίνουμε την εντολή:

$ cd pox

και με την εντολή

$ ./pox.py log.level –DEBUG misc.of_tutorial

προγραμματίζουμε τον ελεγκτή να τρέξει την εφαρμογή. Στη συνέχεια για να επιβεβαιώσουμε την λειτουργία αυτή αλλά και για να δούμε την κίνηση που βλέπει κάθε host από το Mininet δίνουμε την εντολή:

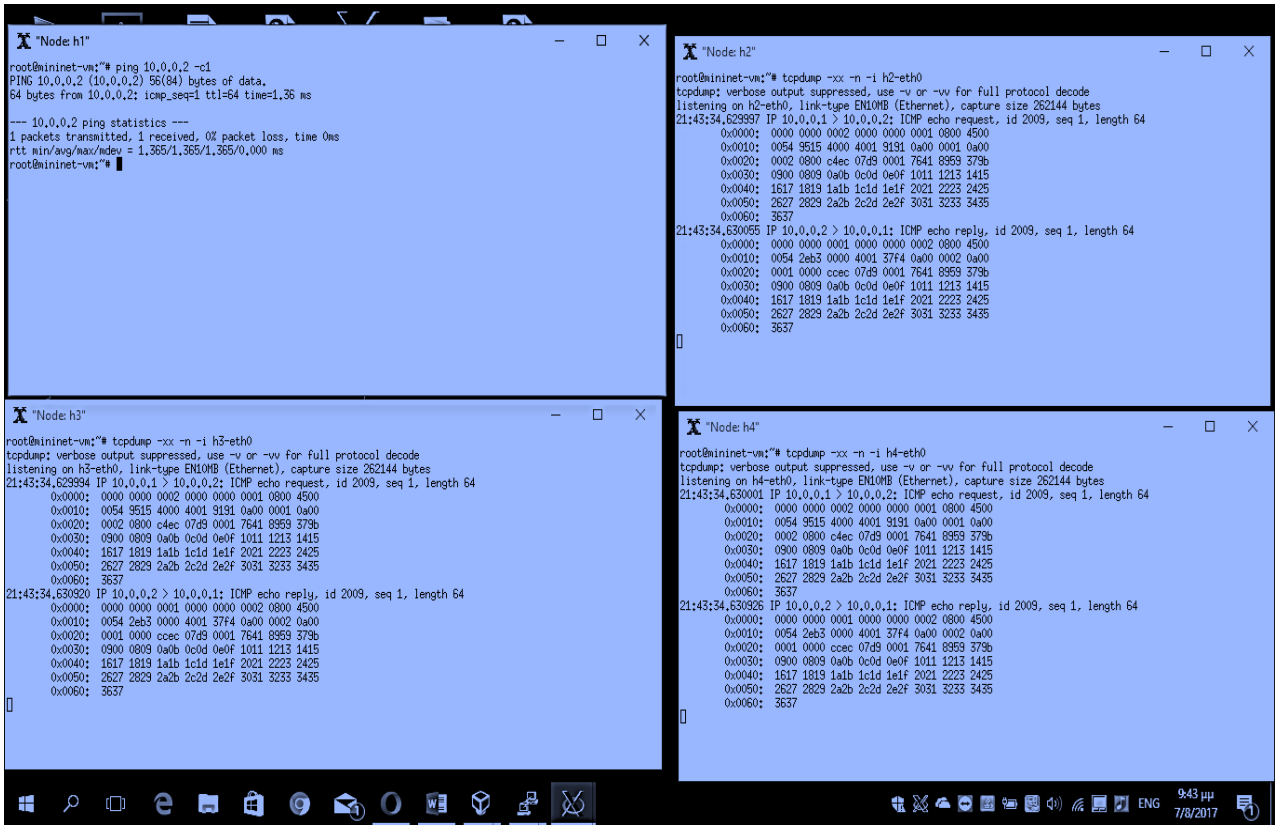
Mininet> *xterm h1 h2 h3 h4*

το οποίο μας ανοίγει τέσσερα παράθυρα για κάθε host της τοπολογίας μας. Σε κάθε host δίνουμε την εντολή *tcpdump* η οποία εμφανίζει τα πακέτα που βλέπει ο κάθε host. Συγκεκριμένα:

tcpdump –xx –n –I h2-eth0, # tcpdump –xx –n –I h3-eth0,# tcpdump –xx –n –I h4-eth0. Ενδεικτική η παρακάτω εικόνα απεικονίζει τις εντολές που μόλις δώσαμε.

Εικόνα 14:Όλοι οι hosts ακούνε για πακέτα εκτός του h1.

Στη συνέχεια, από τον h1 δίνουμε την εντολή: ping –c1 10.0.0.2 (host 2). Αρχικά, το switch μη ξέροντας τι να κάνει με το συγκεκριμένο πακέτο το προωθεί στον ελεγκτή. Ο ελεγκτής ενημερώνει το switch για το πως θα πρέπει να χειριστεί το συγκεκριμένο πακέτο, εγκαθιστώντας στο switch τους κανόνες προώθησης σύμφωνα με το script που τρέχει σε αυτόν. Το αποτέλεσμα είναι όλοι οι hosts να λάβουν ένα αντίγραφο του πακέτου, εξαιτίας του ότι το switch αποστέλλει το πακέτο προς όλους. Όπως φαίνεται ενδεικτικά και από την παρακάτω εικόνα όλοι οι host λαμβάνουν ένα αντίγραφο του πακέτου.



Εικόνα 15:Αποτέλεσμα της λειτουργίας του OpenFlow switch σαν hub, όλοι οι hosts λαμβάνουν και απαντούν στο ping από τον h1.

5.2 Λειτουργία Ενός OpenFlow Switch Σαν L2 Switch

Ένα switch είναι συσκευή επιπέδου 2. Σε αντίθεση με το hub, το οποίο χρησιμοποιεί την μέθοδο της πλημμύρας (flooding) όταν πρόκειται να προωθήσει ένα πακέτο, το switch μαθαίνει τις φυσικές διευθύνσεις σε ένα δίκτυο και βάση αυτών λαμβάνει τις αποφάσεις για την προώθηση ή την απόρριψη ενός πακέτου.

Στη 2η περίπτωση θέλουμε το OpenFlow switch να λειτουργεί σαν ένα L2 switch ενός τωρινού δικτύου. Για να συμβεί αυτό, πρώτα κάνουμε χρήση ενός python script στον POX ελεγκτή. Στη συνέχεια όταν ο ελεγκτής λάβει από το OpenFlow switch αίτημα για το πως θα πρέπει να χειριστεί πακέτα για την συγκεκριμένη τοπολογία που έχουμε δημιουργήσει, του αποστέλλει μέσω του OpenFlow τους κανόνες προώθησης. Αρχικά δημιουργούμε την ίδια ακριβώς τοπολογία όπως στο 5.1 δίνοντας ξανά την εντολή:

$ sudo mn –topo single,4 –mac –switch ovsk –controller remote

Έπειτα, από την κονσόλα του Mininet δίνοντας την εντολή *pingall,* παρατηρούμε ότι κανείς host δεν απαντάει. Ο λόγος είναι ότι δεν έχουμε κάνει ακόμη χρήση του script στον ελεγκτή. Στη συνέχεια, δίνουμε την εντολή στον ελεγκτή να εκτελέσει την εφαρμογή. Πιο συγκεκριμένα:

$ cd pox

$ python. /pox.py forwardin.l2_learning

Έπειτα από τον h1 δίνουμε την εντολή *: # ping –c1 10.0.0.2*

Όπως φαίνεται και στο παρακάτω σχήμα μόνο ο h2 θα λάβει το πακέτο , οι υπόλοιποι (h3, h4) δεν λαμβάνουν τίποτα λόγω της λειτουργίας του OpenFlow switch σαν switch επιπέδου 2 αυτή τη φορά.

Εικόνα 16:Λειτουργία του OpenFlow switch σαν L2 switch, μόνο ο host 1 επικοινωνεί με τον host 2..

5.3 L2 Firewall

Στην τελευταία περίπτωση ο ελεγκτής κάνει χρήση μίας firewall εφαρμογής. Η συγκεκριμένη εφαρμογή αποφασίζει για το ποιοι κόμβοι επιτρέπεται να επικοινωνήσουν κοιτώντας την φυσική διεύθυνση του κάθε κόμβου χρησιμοποιώντας λογική επιπέδου 2. Η συγκεκριμένη εφαρμογή επιτρέπει στον h1 να επικοινωνεί με τον h2 αλλά και το αντίστροφο. Δεν επιτρέπει όμως καμία άλλη επικοινωνία.

Χρησιμοποιούμαι την ίδια τοπολογία με τις δύο προηγούμενες περιπτώσεις, δίνοντας την εντολή:

$ sudo mn –topo single,4 –mac –switch ovsk –controller remote

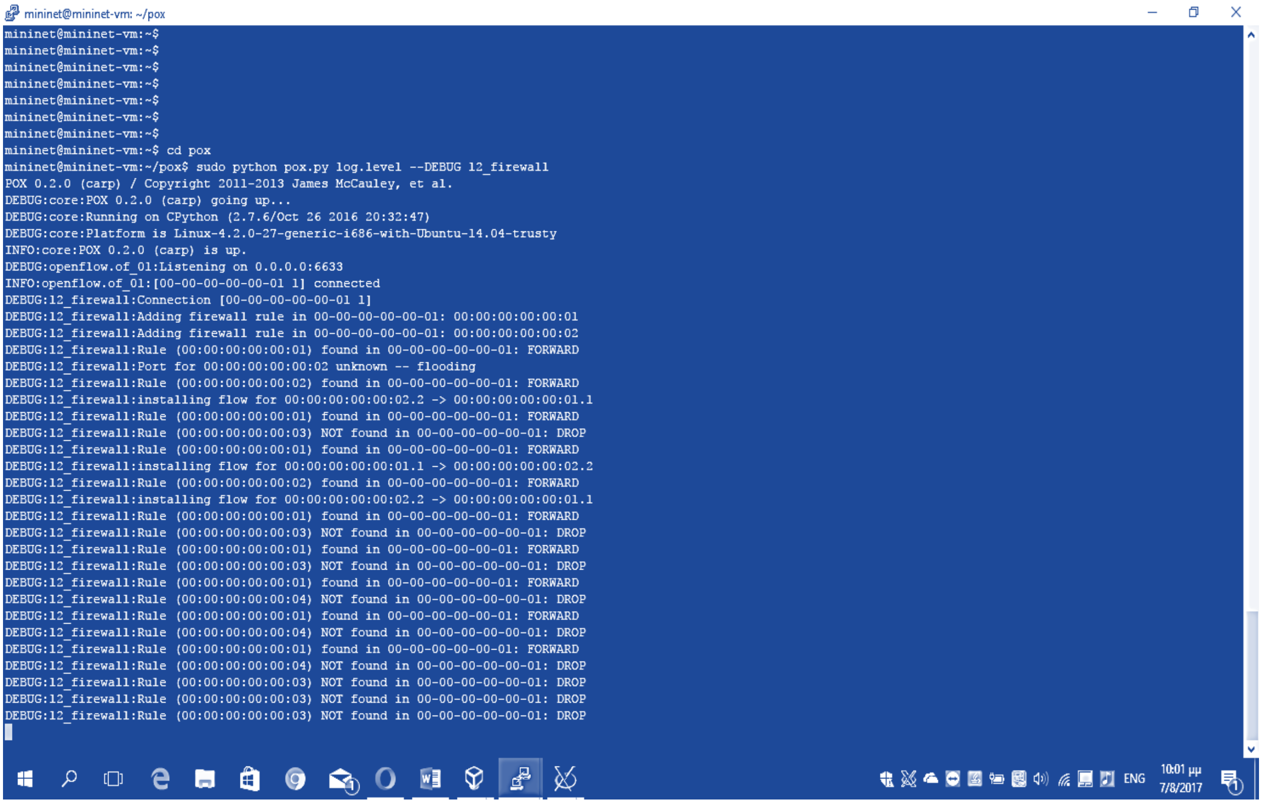
Στη συνέχεια δίνουμε την εντολή στον ελεγκτή να κάνει χρήση του firewall script:

$ cd pox

$ sudo python pox.py log.level –DEBUG l2_firewall

Η τελευταία εντολή εκτελεί το Python script και στην συνέχεια αρχικοποιείται ο ελεγκτής, ο οποίος αρχίζει και ακούει για εισερχόμενα πακέτα χρησιμοποιώντας την θύρα 6633(*DEBUG:openflow.of_01:Listening on 0.0.0:6633*).

Στη συνέχεια επιχειρούμε ping από τον host 1 προς τον host 2. Το switch δεν ξέρει πως να χειριστεί το συγκεκριμένο πακέτο και το στέλνει στον POX. Ο ελεγκτής εξαιτίας του ότι εκτελεί το firewall script, αναγνωρίζει ότι οι δύο συγκεκριμένοι κόμβοι επιτρέπεται να επικοινωνήσουν μεταξύ τους. Αποστέλλει στο switch τον κανόνα προώθησης (*Adding firewall rule*) και το ενημερώνει για την δράση που πρέπει να πάρει για το συγκεκριμένο πακέτο με την εντολή *forward.*Έπειτα επιχειρούμε ping από τον host 3 προς τον host 1. Το switch στέλνει ξανά το πακέτο στον ελεγκτή. Ο ελεγκτής συγκρίνει την φυσική διεύθυνση του host 3 (*firewall rule: 00:00:00:00:00:03*) σύμφωνα με το script που τρέχει και παρατηρεί ότι ο συγκεκριμένος κόμβος δεν έχει δικαίωμα αποστολής πακέτων. Συγκεκριμένα, ο POX αναφέρει με το μήνυμα *NOT found in* ότι δεν υπάρχει κάποια αντιστοίχιση και ενημερώνει το switch ότι η δράση που πρέπει να πάρει για το συγκεκριμένο πακέτο είναι να το απορρίψει (*DROP*).



Εικόνα 17:Εκτέλεση του Python script και αρχικοποίηση του ελεγκτή.

Στη τελευταία εικόνα απεικονίζεται αρχικά το ping του h1 προς τον h2 το οποίο είναι επιτυχές (0% packet loss) σύμφωνα με το python script που τρέχει ο ελεγκτής, και στη συνέχεια τα ping του h1 προς τον h3 και h4 τα οποία είναι ανεπιτυχή (100% packet loss) ακολουθώντας την λογική που τρέχει το script .

Εικόνα 18:Διαδοχικά ping του h1 προς τους h2,h3 και h4.

Κεφάλαιο 6

6.1 Συμπεράσματα

Κλείνοντας, είναι σαφές πως τα SDN δίκτυα διαχωρίζοντας το επίπεδο του ελέγχου από αυτό των δεδομένων παρέχουν μία νέα δυναμική αρχιτεκτονική, η οποία δείχνει έτοιμη να ανταποκριθεί στις τάσεις της αγοράς αλλά και στις ανάγκες τόσο των χρηστών όσο και των επιχειρήσεων. Το SDN υπόσχεται βελτιωμένη αξιοποίηση των δικτυακών πόρων και καλύτερη ποιότητα υπηρεσίας τόσο για ενσύρματα όσο και για ασύρματα δίκτυα σε σχέση με την τωρινή αρχιτεκτονική. Επίσης, με την χρήση του SDN φαίνεται να μειώνονται τόσο οι κεφαλικές όσο και οι λειτουργικές δαπάνες μίας επιχείρησης.

Το OpenFlow πρωτόκολλο είναι το API που επιτρέπει την επικοινωνία μεταξύ του ελεγκτή και των switch με σκοπό τον έλεγχο και την διαχείριση της εκάστοτε συσκευής μέσω μίας γλώσσας υψηλού επιπέδου. Το OpenFlow δίνει την δυνατότητα σε προγραμματιστές, σε ερευνητές και ανεξάρτητους χρήστες να αναπτύξουν νέες δικτυακές τεχνολογίες. Ακόμη, το συγκεκριμένο πρωτόκολλο επιτρέπει την δημιουργία νέων μεθόδων δικτύωσης σε ήδη υπάρχοντα και λειτουργικά δίκτυα, χωρίς να παρεμβαίνει στην λειτουργία των ήδη ενεργών πρωτοκόλλων δρομολόγησης και ασφάλειας. Το αποτέλεσμα είναι το δίκτυο να μετατρέπεται σε ένα πλήρες προγραμματιζόμενο περιβάλλον, βελτιώνοντας την ευελιξία την κλιμάκωση και την διαχείριση του δικτύου.

Χρησιμοποιώντας το OpenFlow σε κινητά και ασύρματα δίκτυα έχουμε την δυνατότητα να αντιμετωπίσουμε τις προκλήσεις που παρουσιάζονται σε τέτοιου είδους περιβάλλοντα, όπως είναι η διαχείριση της παρεμβολής μεταξύ των κυψελών αλλά και την διαχείριση της κίνησης σε κινητά και ασύρματα δίκτυα όπως παρουσιάστηκε και νωρίτερα.

Στη συνέχεια παρουσιάστηκε μία τεχνική (DPT) για το πεδίο των δεδομένων η οποία μπορεί να αποτελέσει ένα χρήσιμο εργαλείο για δικτυακές εφαρμογές, μειώνοντας έτσι τον όγκο κίνησης για τον ελεγκτή. Σύμφωνα με την συγκεκριμένη τεχνική, ένα switch εισόδου προσθέτει ένα timestamp πεδίο σε κάθε πακέτο που εισέρχεται σε ένα SDN δίκτυο το οποίο αφαιρείται από το switch εξόδου, συνεπώς το switch μπορεί να λάβει αποφάσεις αντιστοίχισης και να προβεί στις αντίστοιχες δράσεις σύμφωνα με την τιμή που έχει το timestamp πεδίο του πακέτου.

Τέλος, στο 5^{ο} κεφάλαιο έγινε η προσομοίωση ενός SDN δικτύου και της επικοινωνίας ενός POX ελεγκτή με ένα OpenFlow switch. Σαν εργαλείο προσομοίωσης χρησιμοποιήθηκε το Mininet το οποίο εγκαταστάθηκε σαν εικονική μηχανή στο Virtual Box. Μελετήθηκαν τρία διαφορετικά σενάρια προσομοίωσης. Αρχικά, μέσω του POX ελεγκτή διαμορφώσαμε το OpenFlow switch να λειτουργεί σαν μία hub συσκευή, στη συνέχεια το switch διαμορφώθηκε ώστε να λειτουργεί σαν συσκευή επιπέδου 2. Τέλος, στην τελευταία περίπτωση έγινε χρήση μίας firewall εφαρμογής στον ελεγκτή. Η συγκεκριμένη εφαρμογή στην συνέχεια διαμορφώθηκε στο switch

μέσω του OpenFlow, παρατηρήθηκε πως μόνοι οι κόμβοι που είχαν την δυνατότητα να επικοινωνούν μεταξύ τους, ήταν εκείνοι που είχαμε ορίσει στο script που εκτελέστηκε στον POX.

6.2 SDN Δίκτυα Και Προκλήσεις

Παρόλο που τα SDN δίκτυα προσφέρουν σημαντικές λύσεις που αφορούν τον σχεδιασμό και την διαχείριση ενός δικτύου, η εφαρμογή του σε ένα πραγματικό περιβάλλον κρύβει προκλήσεις και παγίδες που πρέπει να ξεπεραστούν **[22].** Στη συνέχεια παρουσιάζονται κάποιες από τις προκλήσεις που πρέπει να ξεπεράσουν τα SDN δίκτυα ώστε να τύχουν ευρείας αποδοχής.

6.2.1 Ζητήματα Διαλειτουργικότητας

Μία από τις βασικότερες προκλήσεις για την επιτυχή εφαρμογή του SDN, είναι η διασφάλιση της διαλειτουργικότητας (interoperability) με τα υπάρχοντα δίκτυα. Αυτή τη στιγμή, οι επιχειρήσεις βασίζονται στα παραδοσιακά δίκτυα για την λειτουργία κρίσιμων εφαρμογών **[22]** . Έτσι, η αντικατάσταση ενός παραδοσιακού δικτύου με ένα SDN δίκτυο δεν είναι δυνατό να συμβεί σε μία ημέρα ή και εβδομάδα. Το πιθανότερο που ενδέχεται να συμβεί από μία τέτοια άμεση μετάβαση είναι διακοπές στο δίκτυο και πελάτες που δεν θα μπορούν να εξυπηρετηθούν (Denial of Service). Για το λόγο αυτό, είναι σημαντικό η εφαρμογή του SDN να συμβεί σταδιακά. Ο οργανισμός IETF (Internet Engineering Task Force), εργάζεται για την ανάπτυξη προτύπων για πρωτόκολλα, για διεπαφές αλλά και για μηχανισμούς ώστε να διασφαλιστεί η ομαλή λειτουργία του SDN με τα τωρινά δίκτυα.

6.2.2 Ζητήματα Ασφάλειας

Με την υιοθέτηση των SDN δικτύων και την εισαγωγή νέων χαρακτηριστικών και υλοποιήσεων, ανοίγει ο δρόμος για καινούριες απειλές ασφάλειας, οι οποίες στα παραδοσιακά δίκτυα δεν υπήρχαν. Στην συνέχεια παραθέτονται μερικές απειλές ασφάλειας που μπορούν να επηρεάσουν την λειτουργία του SDN.

- **Άρνηση εξυπηρέτησης (Denial of Service)**
  Ο επιτιθέμενος μπορεί να ωθήσει προς τον ελεγκτή μεγάλο όγκο κίνησης, η οποία θα αλλάζει συνεχώς και με τυχαίο τρόπο τα χαρακτηριστικά των ροών. Έτσι, ο ελεγκτής λαμβάνει καινούριες και άγνωστες ροές κίνησης τις οποίες δεν γνωρίζει πως να τις χειριστεί. Το αποτέλεσμα είναι διακοπές στο δίκτυο και άρνηση εξυπηρέτησης.
- **Ευπάθειες του switch (Switch vulnerabilities)**
  Ένα κακόβουλο switch, μπορεί να πλημμυρίσει τον ελεγκτή με κακόβουλα πακέτα. Το αποτέλεσμα θα είναι η επιβράδυνση της δικτυακής κίνησης ή ακόμη ο ελεγκτής να μην απαντάει σε καινούρια αιτήματα δρομολόγησης από άλλα switch.

- **Man-in-the-middle Attacks**
  Αν δεν γίνεται χρήση του TLS πρωτοκόλλου για την επικοινωνία μεταξύ του ελεγκτή και του switch, ο επιτιθέμενος μπορεί να εκμεταλλευθεί τις ευπάθειες του επιπέδου ελέγχου και να πραγματοποιήσει από εκεί επιθέσεις.
- **Ζητήματα Αυθεντικοποίησης (Authentication Issues)**
  Σε ένα SDN δίκτυο, πολλαπλοί κόμβοι έχουν πρόσβαση σε ένα ελεγκτή, έτσι το ενδεχόμενο της μη-εξουσιοδοτημένης πρόσβασης και μη-νόμιμης διαμόρφωσης αυξάνει.
- **Ανοιχτές Διεπαφές (Open Interfaces)**
  Δεδομένου ότι το SDN υποστηρίζει ανοιχτές διεπαφές και γνωστά πρωτόκολλα με σκοπό την απλοποίηση της δικτύωσης, ένας επιτιθέμενος μπορεί να το εκμεταλλευτεί και να αποκτήσει πλήρη έλεγχο του δικτύου.

6.2.3 Ζητήματα Κλιμάκωσης (Scalability Issues)

Όπως έχουμε αναφέρει και στα προηγούμενα κεφάλαια, ένα από τα βασικότερα χαρακτηριστικά των SDN δικτύων είναι η αφαίρεση του επιπέδου των δεδομένων από το επίπεδο ελέγχου της κάθε συσκευής. Έτσι, η προσθήκη νέων συσκευών σε ένα δίκτυο με σκοπό την ικανοποίηση των απαιτήσεων για κλιμάκωση, δεν αλλάζει τις λειτουργίες ελέγχου του δικτύου, προσφέροντας παράλληλα αδιαφανή κλιμάκωση **[23].**

Παρόλο που η προσθήκη νέων συσκευών σε ένα SDN δίκτυο είναι αδιαφανής, υπάρχουν κάποια ζητήματα κλιμάκωσης που πρέπει να αντιμετωπιστούν. Πρώτον, η αύξηση των συσκευών και των ροών, έχει σαν αποτέλεσμα τον πολλαπλασιασμό της κίνησης, την οποία ενδεχομένως ο ελεγκτής να μην μπορεί να εξυπηρετήσει **[22].** Έτσι, σε θέματα που αφορούν την επικοινωνία μεταξύ του ελεγκτή και της εκάστοτε συσκευής, η συσκευή ενδέχεται να περιμένει μεγαλύτερο χρονικό διάστημα για την απάντηση του ελεγκτή. Επίσης, όπως αναφέραμε και στο κεφάλαιο 2, η προσθήκη μίας νέας ροής, απαιτεί την δημιουργία μίας νέας καταχώρησης στον πίνακα ροής της συσκευής. Το switch από την πλευρά του πρέπει να πραγματοποιήσει μία διαδικασία ενημέρωσης, η οποία αφορά την προσθήκη της νέας ροής. Επιπλέον, υπάρχουν περιπτώσεις όπου η συσκευή πρέπει να επικοινωνήσει με τον ελεγκτή για την δημιουργία μίας νέας καταχώρησης. Έτσι, με την συνεχή προσθήκη συσκευών σε ένα δίκτυο, αυξάνονται οι ενημερώσεις που πρέπει να πραγματοποιήσουν οι συσκευές, με αποτέλεσμα την περαιτέρω επιβάρυνση του ελεγκτή. Μία από τις λύσεις που έχει προταθεί για την επίλυση του ζητήματος είναι, οι συσκευές να επεξεργάζονται ένα μέρος των αιτημάτων από μόνες τους, έτσι ώστε ο ελεγκτής να μην κατακλύζεται από χιλιάδες αιτήματα **[23]**.

Επίσης η συγκέντρωση όλων των λειτουργιών σε ένα μοναδικό κόμβο, απαιτεί μεγάλη επεξεργαστική ισχύ, αποθηκευτικότητα και δυνατότητα διακίνησης δεδομένων, οι οποίες με την σειρά τους αυξάνουν τον χρόνο απόκρισης. Επίσης, λόγω των περιορισμών του υλικού των switches, αυξάνεται η πιθανότητα

δημιουργίας σημείων συμφόρησης στο δίκτυο. Επίσης, σε μεγάλου μεγέθους datacenters και δίκτυα υπολογιστικής cloud, οι λύσεις για τα προβλήματα κλιμάκωσης δεν είναι ικανές να επιφέρουν τα επιθυμητά αποτελέσματα.

6.2.4 Διαθεσιμότητα Της Υπηρεσίας

Στα παραδοσιακά δίκτυα, όταν μία ή και περισσότερες συσκευές για οποιοδήποτε λόγο δεν είναι διαθέσιμες, η κίνηση μπορεί να δρομολογηθεί μέσω εναλλακτικών διαδρομών, ώστε να διασφαλιστεί η συνεχής παροχή των υπηρεσιών. Τα SDN δίκτυα, εξαιτίας της αρχιτεκτονικής τους, εμπεριέχουν ένα ζήτημα που ονομάζεται μοναδικό σημείο αποτυχίας (single point of failure) **[24]**. Για παράδειγμα, όταν σε ένα SDN δίκτυο υπάρχει μόνο ένας ελεγκτής διαθέσιμος για την εξυπηρέτηση των χρηστών, και ο ελεγκτής για κάποιο λόγο καταστεί αδύνατο να λειτουργήσει, τότε το αποτέλεσμα θα είναι να μην μπορεί να εξυπηρετηθεί κανένας χρήστης **[24]**. Ακόμη, αρκεί να σκεφτούμε την περίπτωση που κάποιος κακόβουλος χρήστης έχει τον έλεγχο του ελεγκτή, το αποτέλεσμα θα είναι η διατάραξη της λειτουργίας του δικτύου. Συνεπώς, είναι σημαντικό να υπάρχει τουλάχιστον ένας ακόμη ελεγκτής σαν εφεδρική λύση του υπάρχοντος ελεγκτή. Επίσης, οι αποτυχίες συνδέσεων παρεμποδίζουν την διαθεσιμότητα μίας υπηρεσίας που προορίζεται για τους χρήστες. Έτσι, η SDN αρχιτεκτονική πρέπει να υποστηρίζει την διαμόρφωση πολλαπλών διαδρομών, ώστε ο ελεγκτής να μπορεί να κατευθύνει την κίνηση από μη-ενεργές συνδέσεις σε ενεργές **[24]**.

6.2.5 Ζητήματα Απόδοσης (Performance Issues)

Όπως έχουμε αναφέρει ήδη, εξαιτίας της αρχιτεκτονικής των SDN δικτύων, υπάρχουν περιπτώσεις όπου ο ελεγκτής κατακλύζεται από μεγάλο αριθμό αιτημάτων. Το αποτέλεσμα είναι είτε κάποια αιτήματα να καθυστερεί να τα εξυπηρετήσει, είτε κάποια άλλα να μην τα εξυπηρετεί καθόλου **[25].** Μια λύση που έχει προταθεί για το συγκεκριμένο πρόβλημα περιλαμβάνει, μία υβριδική SDN αρχιτεκτονική με πολλαπλούς ελεγκτές, οι οποίοι μοιράζονται τον όγκο εργασίας .Μία τέτοια υβριδική αρχιτεκτονική, μπορεί να διασφαλίσει την γρήγορη προώθηση των ροών **[26].**

6.2.6 Συχνή Ανανέωση Των OpenFlow Switch

Ο ελεγκτής θα πρέπει να ανανεώνει τα OpenFlow switch συχνότερα από ότι απαιτούν τα παραδοσιακά switch. Συνεπώς, υπάρχει μεγαλύτερη επιβάρυνση για το δίκτυο. Επίσης, οι ροές ταξινομούνται σε κατηγορίες, με διαφορετική προτεραιότητα και διαφορετικές απαιτήσεις σε QoS. Το αποτέλεσμα είναι να δημιουργούνται θέματα ευελιξίας, κυρίως σε δίκτυα ευρείας κλίμακα

Παράρτημα Κώδικα Της Firewall Εφαρμογής

```
from pox.core import core
import pox.openflow.libopenflow_01 as of
from pox.lib.util import dpid_to_str
from pox.lib.util import str_to_bool
import time
from pox.lib.addresses import EthAddr

log = core.getLogger()
_flood_delay = 0


class LearningSwitch (object):
      def __init__ (self, connection, transparent):
      # Switch we'll be adding L2 learning switch capabilities to
      self.connection = connection
      self.transparent = transparent

       # Our table
       self.macToPort = { }
      # Our firewall table
       self.firewall = { }

# Add a Couple of Rules
self.AddRule('00-00-00-00-00-01',EthAddr('00:00:00:00:00:01'))
self.AddRule('00-00-00-00-00-01',EthAddr('00:00:00:00:00:02'))

# We want to hear PacketIn messages, so we listen
# to the connection
connection.addListeners(self)

# We just use this to know when to log a helpful message
self.hold_down_expired = _flood_delay == 0


# function that allows adding firewall rules into the firewall table

def AddRule (self, dpidstr, src=0,value=True):
   self.firewall [(dpidstr,src)]=value
   log.debug("Adding firewall rule in %s: %s", dpidstr, src)


# function that allows deleting firewall rules from the firewall table
  def DeleteRule (self, dpidstr, src=0):
    try:
      del self.firewal[(ldpidstr,src)]
      log.debug("Deleting firewall rule in %s: %s", dpidstr, src)
```

```
    except KeyError:
     log.error("Cannot find in %s: %s", dpidstr, src)


# check if packet is compliant to rules before proceeding

def CheckRule (self, dpidstr, src=0):
   try:
    entry = self.firewall [(dpidstr, src)]
    if (entry == True):
     log.debug("Rule (%s) found in %s: FORWARD", src, dpidstr)
    else:
     log.debug("Rule (%s) found in %s: DROP", src, dpidstr)
    return entry
   except KeyError:
    log.debug("Rule (%s) NOT found in %s: DROP", src, dpidstr)
    return False
def _handle_PacketIn(self, event):
 """
    Handle packet in messages from the switch to implement
above algorithm.
    """
  packet = event.parsed

def flood (message = None):
  """ Floods the packet """
  msg = of.ofp_packet_out()
  if time.time() - self.connection.connect_time >=
_flood_delay:
# Only flood if we've been connected for a little while...

      if self.hold_down_expired is False:
           # Oh yes it is!
           self.hold_down_expired = True
           log.info("%s: Flood hold-down expired -- flooding",
               dpid_to_str(event.dpid))

       if message is not None: log.debug(message)

      msg.actions.append(of.ofp_action_output(port
of.OFPP_FLOOD))

  else:
        pass
        #log.info("Holding down flood for %s",
dpid_to_str(event.dpid))

  msg.data = event.ofp
  msg.in_port = event.port
  self.connection.send(msg)
```

```
def drop (duration = None):
      """
      Drops this packet and optionally installs a flow to
continue
      dropping similar ones for a while
      """
      if duration is not None:
        if not isinstance(duration, tuple):
          duration = (duration,duration)
        msg = of.ofp_flow_mod()
        msg.match = of.ofp_match.from_packet(packet)
        msg.idle_timeout = duration[0]
        msg.hard_timeout = duration[1]
        msg.buffer_id = event.ofp.buffer_id
        self.connection.send(msg)
      elif event.ofp.buffer_id is not None:
        msg = of.ofp_packet_out()
        msg.buffer_id = event.ofp.buffer_id
        msg.in_port = event.port
        self.connection.send(msg)

self.macToPort[packet.src] = event.port # 1

# Get the DPID of the Switch Connection
dpidstr = dpid_to_str(event.connection.dpid)

# Check the Firewall Rules
if self.CheckRule(dpidstr, packet.src) == False:
  drop()
  return

if not self.transparent: # 2
   if packet.type == packet.LLDP_TYPE or
packet.dst.isBridgeFiltered():
   drop() # 2a
   return

if packet.dst.is_multicast:
      flood() # 3a
else:
  if packet.dst not in self.macToPort: # 4
    flood("Port for %s unknown -- flooding" % (packet.dst,))

  else:
    port = self.macToPort[packet.dst]
    if port == event.port: # 5
          # 5a
       log.warning("Same port for packet from %s -> %s on
%s.%s.  Drop." % (packet.src, packet.dst,
dpid_to_str(event.dpid), port))
       drop(10)
```

```
        return


    # 6
    log.debug("installing flow for %s.%i -> %s.%i" %
              (packet.src, event.port, packet.dst, port))
    msg = of.ofp_flow_mod()
    msg.match = of.ofp_match.from_packet(packet, event.port)
    msg.idle_timeout = 10
    msg.hard_timeout = 30
    msg.actions.append(of.ofp_action_output(port = port))
    msg.data = event.ofp # 6a
    self.connection.send(msg)


class l2_learning (object):
  """
  Waits for OpenFlow switches to connect and makes them
learning switches.
  """
  def __init__ (self, transparent):
    core.openflow.addListeners(self)
    self.transparent = transparent

  def _handle_ConnectionUp (self, event):
    log.debug("Connection %s" % (event.connection,))
    LearningSwitch(event.connection, self.transparent)


def launch (transparent=False, hold_down=_flood_delay):
  """
  Starts an L2 learning switch.
  """
  try:
    global _flood_delay
    _flood_delay = int(str(hold_down), 10)
    assert _flood_delay >= 0
  except:
    raise RuntimeError("Expected hold-down to be a number")

  core.registerNew(l2_learning, str_to_bool(transparent))
```

Κεφάλαιο 7
Βιβλιογραφία

[1] ONF, https://www.opennetworking.org

[2] ONF, Software-Dened Networking: The New Norm for Networks.

[3] OpenFlow Specifications Sheet v1.3.3 2013.

[4] D. Kreutz et al., Software defined networking: A comprehensive Survey, Proc. IEEE, vol. 103, no. 1, pp. 14–76, Jan. 2015.

[5] ONF, "OpenFlow-Enabled Mobile and Wireless Networks", 2013.

[6] Wireless Software Defined Networks: Challenges and Opportunities, Claude Chaudet Telecom ParisTech, Institut Telecom Paris, Yoram Haddad Jerusalem College of Technology Jerusalem.

[7] Wireless Software Defined Networking: A Survey and Taxonomy, Israat Tanzeena Haque, Senior Member, IEEE, and Nael Abu-Ghazaleh, Senior Member, IEEE.

[8] Control and provisioning of wireless access points (capwap) protocol specification. RFC 5415, Mar. 2009.

[9] K.-K. Yap, S. Katti, G. Parulkar, and N. McKeown. Delivering capacity for the mobile internet by stitching together networks. In ACM workshop on Wireless of the students, by the students, for the students, 2010.

[10] Aditya Gudipati, Daniel Perry, Li Erran Li, Sachin Katti, SoftRAN: Software Defined Radio Access Network, HotSDN 13, August 16, 2013, Hong Kong, China

[11] Jin X., Li L. E., Vanbever L., Rexford J. SoftCell: Scalable and Flexible Cellular Core Network Architecture. InProceedings of ACMCoNEXT. 2013

[12] CellSDN: Software-Defined Cellular Networks, Li Erran Li Bell Labs, Z. Morley Mao University of Michigan, Jennifer Rexford Princeton University.

[13] TIMEFLIP: Scheduling Network Updates with Timestamp-based TCAM Ranges, Tal Mizrahi Technion, Ori Rottenstreich Princeton University, Yoram Moses Technion.

[14] Time-based Updates in Software Defined Networks, Tal Mizrahi Technion — Israel Institute of Technology, Yoram Moses Technion — Israel Institute of Technology.

[15] Time and Timestamping in Softwarized Environments, Tal Mizrahi, Yoram Moses Technion – Israel Institute of Technology, 2016.

[16] Mininet: An Instant Virtual Network on your Laptop. http://mininet.org/.

[17] POX, Pox openflow controller, 2014, Accessed: Sept.2014. [Online]: https://openflow.stanford.edu/dashboard.action .

[18] http://www.ieee802.org/3/

[19] http://www.ieee802.org/11/

[20] V. Jacobson, B. Braden, and D. Borman, "TCP extensions for high performance," RFC 1323, IETF, 1992.

[21] C. Kim, P. Bhide, E. Doe, H. Holbrook, A. Ghanwani, D. Daly, M. Hira, and B. Davie, "In-band network telemetry (INT)," technical specification, P4, 2015.

[22] S. Sezer, S. Scott-Hayward, P. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, N. Rao, "Are we ready for SDN? Implementation challenges for software-defined networks", IEEE Communications Magazine, vol. 51, no. 7, pp. 36-43, 2013.

[23] S. Yeganeh, A. Tootoonchian, Y. Ganjali, "On scalability of software-defined networking," IEEE Communications Magazine, vol. 51, no. 2, pp. 136-141, 2013.

[24] Ashton, Metzler, and Associates, "Ten Things to Look for in an SDN Controller", Technical Report, 2013.

[25] A. Tootoonchian, S. Gorbunov, "On Controller Performance in Software-Defined Networks," In Proceedings of 2nd USENIX Conference on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services, vol. 54, pp. 10, 2012.

[26] J. Mogul, L. Tourrilhes, P. Yalagandula, P. Sharma, A. Curtis, S. Banerjee, "DevoFlow: Cost-Effective Flow Management for High-Performance Enterprise Networks," In Proceedings of the Ninth ACM SIGCOMM Workshop on Hot Topics in Networks(HotnetsIX), pp. 1, 2010.

[27] Akram Hakiri, Pascal Berthou,"Leveraging SDN for The 5G Networks: Trends, Prospects and Challenges", arXiv:1506.02876, June 2015.

[28] Ian F. Akyildiz, Pu Wang, Shih-Chun Lin,"SoftAir: A software defined networking architecture for 5G wireless systems", Computer Networkins, Elsevier, June 2015.

[29] Guolin Sun, Feng Liu, Junyu Lai, and Guisong Liu, "Software Defined Wireless Network Architecture for the Next Generation Mobile Communication, December 2014.

[30] ND Szabo, F Nemeth, B Sonkoly, A Gulyas, FHP Fitzek, " Towards the 5G Revolution: A Software Defined Network Architecture Exploiting Network Coding as a Service",