Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Ενσωμάτωση της SDN αρχιτεκτονικής στο LTE δίκτυο** **Integration of SDN architecture in LTE networks**
Ονοματεπώνυμο Φοιτητή	**Θοδωρής Καρκασίνας**
Πατρώνυμο	**Βασίλης**
Αριθμός Μητρώου	**ΜΠΠΛ 13034**
Επιβλέπων	**Δημήτριος Βέργαδος, Αναπληρωτής Καθηγητής**

Ημερομηνία Παράδοσης **Ιούλιος 2016**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
Όνομα Επώνυμο Βαθμίδα	Όνομα Επώνυμο Βαθμίδα	Όνομα Επώνυμο Βαθμίδα

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Αναγνώριση

Η εργασία αυτή αποτελεί το επιστέγασμα των γνώσεων που απέκτησα απο το Μεταπτυχιακό πρόγραμμα. Θα ήθελα να ευχαριστήσω το ακαδημαϊκό προσωπικό, τους επιτηρητές και την επιτροπή για την συνεργασία κατα την διάρκεια των σπουδών μου.

Ιδιαίτερα θα ήθελα να ευχαρηστήσω τον επιτηρητή και Αναπληρωτή Καθηγητή κ. Βέργαδο Δημήτριο που με βοήθησε να αναπτύξω τις ερευνητικές μου ικανότητες και να διευρύνω τον τρόπο σκέψης μου. Επίσης, θα ήθελα να ευχαριστήσω τον Αναπληρωτή Καθηγητή κ. Άγγελο Μιχάλα για την υποστήριξη στην ανάπτυξη της εργασίας, στις καίριες παρατηρήσεις του, καθώς και στην συνολικότερη επιτήρηση και αποφυγή των αδιεξόδων, όπως και την κ. Αγγελική Σγώρα για την συνεργασία.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς και τα αδέρφια μου για την στήριξη και υπομονή τους.

Περίληψη

Στα πλαίσια της παρούσας μεταπτυχιακής διατριβής μελετήθηκε η εφαρμογή του SDN (Software Defined Networks) και NFV (Network Functions Virtualization) σε δίκτυα LTE.

Στο πρώτο κεφάλαιο παρουσιάζεται η εξέλιξη των τηλεπικοινωνιακών δικτύων απο την πρώτη γενιά μέχρι την την πέμπτη που ακόμα δεν έχει εμπορική εφαρμογή. Στο δεύτερο κεφάλαιο γίνεται αναλυτική περιγραφή του τρόπου λειτουργίας του SDN και του πρωτοκόλλου OpenFlow. Στο τρίτο κεφάλαιο γίνεται παρουσίαση προτάσεων εφαρμογής του SDN σε δίκτυα LTE, προβλημάτων της υπάρχουσας 3GPP αρχιτεκτονικής, προτεινόμενες λύσεις με την χρήση του SDN και προτείνεται μια αρχιτεκτονική ενος LTE δικτύου βασισμένο στο SDN. Στο τέταρτο κεφάλαιο παρουσιάζονται τα πειραματικά αποτελέσματα της αρχιτεκτονικής που προτάθηκε.

Τα αποτελέσματα αφορούν την δυναμική διαχείριση του δικτύου και το διαμοιρασμό του φόρτου.

ABSTRACT

In this thesis we study the implementation of Software Defined Networks in existing LTE architecture.

SDN is a way to decouple control plane from user plane in networking. To achieve that new hardware and software need to be introduced. The main newly entered piece of hardware is the OpenFlow (OF) controller, which is responsible for converting a network from hardware defined to software defined. OF protocol is also essential for SDN networks to exist. Also OF switches are important parts of reshaping the design of networks, either as virtual switches or as physical.

SDN could give cellular operators greater control over their equipment, simplify network management, and introduce value-added services. SDN can enable carriers to distribute data-plane rules over multiple, cheaper network switches, reducing the scalability pressure on the packet gateway and enabling flexible handling of traffic that stays within the cellular network

Λίστα Συντομογραφιών

1G: 1st Generation

2G: 2nd Generation

3G: 3d Generation

4G: 4th Generation

5G: 5th Generation

AMPS: Advanced Mobile Phone Service

NMT: Nordic Mobile Telephone

TACS: Total Access Communication System

GSM: Global System for Mobile Communications

CDMA: Code Division Multiple Access

WCDMA: Wideband CDMA

3GPP: Third Generation Partnership Project

PDC: Personal Digital Cellular

JDC: Japanese Digital Cellular

HSCSD: High-Speed Circuit-Switched Data

GPRS: General Packet Radio Services

TDMA: Time Division Multiple Access

EDGE: Enhanced Data Rates for Global Evolution

8-PSK: 8-Phase Shift Keying

GMSK: Gaussian Minimum Shift Keying

EGPRS: Enhanced GPRS

UE: User Equipment

USIM: UMTS Service Identity Module

UTRAN: Universal Terestrial Radio Access Network

BTS: Base Transceiver Station

BSC: Base Station Controller (BSC)

RAN: Radio Access Network

CN: Core Network

MIMO: Multiple Input/Multiple Output

HRPD: High Rate Packet Data

HSDPA: High-Speed Downlink Packet Access

PAPR: Peak-to-Average Power Ratio

TDD: Time Division Duplexing

FDD: Frequency Division Duplexing

WiFi: Wireless Fidelity

WiMAX: Worldwide Interoperability for Microwave Access

AMC: Adaptive Modulation and Coding

HSUPA: High-Speed UplinkPacket Access

ARQ: Automatic Retransmission Requests

Carrier Sense Multiple Access (CSMA)

RNC: Radio Network Controller

SDN: Software-Defined Networking

NFV: Network Functions Virtualization

ONF : Open Networking Foundation

CPU: Central Prossesing Unit

GPU: General Processing Unit

NPU: Network Proccesing Units

NFP: Network Flow Processors

SON: Self-Organizing Networks

PLD: Programmable Logical Devices

TLS: Transport Layer Security

ETSI ISG NFV: ETSI Industry Specification Group for Network Functions Virtualization

OXM: OpenFlow Extensible Match

TLV: Type Length Value

DNS: Domain Name System

PCE: Path Computation Element

RAT: Radio Access Transfer

TA: Tracking Area

VoLTE: Voice over LTE

API: Application Programming Interface

MME: Mobility Management Entity

SGW: Serving Gateway

PGW: PDN Gateway

HSS: Home Subscriber Server

SDMN: Software Defined Mobile Network

TEID: Tunnel Endpoint Identifier

IP: Internet Protocol

QoS: Quality of Service

UDP: User Datagram Protocol

GTP: GPRS Tunneling Protocol

TCP: Transmission Control Protocol

VoIP: Voice over IP

DiffServ: Differentiated Services

ToS: Type of Service

NGNI: Next Generation Network Infrastructure

HTTP: Hypertext Transfer Protocol

AF: Assured Forwarding

BE: Best Effort

EF: Expedited Forwarding

DSCP: Differentiated Services Code Point

Κεφάλαιο 1°

1.1 Κινητά Δίκτυα 1η Γενιά

Η ανάπτυξη των κινητών δικτύων μπορεί να διακριθεί σε γενιές. Τα πρώτα συστήματα κινητών τηλεπικοινωνιών δεν είχαν τα χαρακτηριστικά που γνωρίζουμε σήμερα. Σε αυτά τα δίκτυα, που τοποθετούνται χρονολογικά πριν την εμφάνιση της πρώτης γενιάς (1G) κυψελωδών δικτύων, η κινητικότητα ήταν υποτυπώδης και δεν είχαν κυψελώδη δομή [1].

Η εισαγωγή των κυψελωδών δικτύων πρώτης γενιάς (1G) έγινε στα τέλη της δεκαετίας του 1970. Το κύριο χαρακτηριστικό είναι η διαίρεση της περιοχής κάλυψης σε κυψέλες, με αποτέλεσμα οι ίδιες συχνότητες να μπορούν να χρησιμοποιηθούν πολλές φορές στο ίδιο δίκτυο [1]. Οι υπηρεσίες που παρείχαν τα δίκτυα πρώτης γενιάς ήταν αποκλειστικά η φωνή, η μετάδοση της οποίας γίνετε με την χρήση αναλογικών τεχνικών. Το πρότυπο Advanced Mobile Phone Service (AMPS) επικράτησε στις Ηνωμένες Πολιτείες Αμερικής, ενώ στην Ευρώπη τα πρότυπα Nordic Mobile Telephone (NMT) και Total Access Communication System (TACS) είχαν σχετική επιτυχία.

Το πρότυπο NMT χρησιμοποιήθηκε από αρκετές χώρες της Ευρώπης. Υπήρχαν δύο εκδόσεις, οι NMT-450 και NMT-900. Η έκδοση NMT-450 χρησιμοποιούσε συχνότητες των 450 MHz, ενώ το NMT-900 χρησιμοποιούσε συχνότητες στα 900 MHz. Το πρότυπο έδινε την δυνατότητα χρήσης της υπηρεσίας roaming για διεθνείς κλήσεις. Το TACS είναι ένα πρότυπο που αναπτύχθηκε από το Ηνωμένο Βασίλειο και βασίστηκε πάνω στο AMPS. Το TACS χρησιμοποιούσε συχνότητες στα 900 MHz, το AMPS χρησιμοποιούσε συχνότητες στα 800 MHz.

Ενώ τα συστήματα πρώτης γενιάς παρείχαν αρκετά καλή ποιότητα φωνής είχαν κακή αποδοτικότητα φάσματος (spectral efficiency). Αυτός ήταν ο λόγος που οδηγηθήκαμε στην δημιουργία συστημάτων δεύτερης γενιάς (2G) [2].

1.2 Γενιά 2

Στη δεύτερη γενιά-2G- κινητών τηλεπικοινωνιακών δικτύων χρησιμοποιούνται ψηφιακές τεχνικές μετάδοσης. Τα δύο ευρέως διαδεδομένα κυψελωτά συστήματα 2ης γενιάς είναι το GSM (Global System for Mobile Communications) και το CDMA (Code Division Multiple Access), το οποίο ήταν γνωστό ως το Αμερικάνικο interim standard 95 ή IS-95, τώρα καλείται ως cdmaOne [3].

Το GSM είναι το πιο επιτυχημένο σύστημα 2ης γενιάς, το οποίο ξεκίνησε αρχικά ως ευρωπαϊκό σύστημα αλλά επικράτησε και σε παγκόσμιο επίπεδο. Στην αμερικάνικη ήπειρο η κοινότητα TDMA (Time Division Multiple Access) ως 3G τεχνολογία, υιοθετήθηκε το WCDMA (Wideband CDMA), του οποίου οι προδιαγραφές καθορίζονται από το 3GPP (Third Generation Partnership Project). Πολλές αμερικάνικες

εταιρίες, που χρησιμοποιούσαν το Digital AMPS (D-AMPS), προκειμένου να προετοιμαστούν για το WCDMA επέλεξαν το σύστημα GSM/GPRS [1].

Το βασικό GSM σύστημα χρησιμοποιεί το εύρος των 900MHz, αλλά υπάρχουν πολλά παράγωγα όπως το GSM-1800 και το GSM-1900. Τα πρότυπα αυτά χρησιμοποιούν το εύρος συχνοτήτων των 1800 MHz και 1900 MHz αντίστοιχα. Ο λόγος για τον οποίο χρησιμοποιούνται αυτές οι συχνότητες είναι η έλλειψη χωρητικότητας στο εύρος των 900 MHz. Επειδή το εύρος των 1800 MHz μπορεί να εξυπηρετήσει μεγαλύτερο αριθμό χρηστών έχει γίνει αρκετά δημοφιλές ειδικότερα σε πυκνοκατοικημένες περιοχές. Καλύπτει, όμως, μικρότερη έκταση απ' ότι το εύρος των 900 MHz. Σε αυτήν την περίπτωση, χρησιμοποιούνται συσκευές dual band, οι οποίες μπορούν να λειτουργούν τόσο στα 900MHz όσο και στα 1800MHz.

Το πρότυπο CDMA αναπτύχθηκε από την Qualcomm και χρησιμοποιεί διαφορετικό σχεδιασμό για τη διεπαφή επικοινωνίας . Επιπλέον, χρησιμοποιεί διαφορετικούς κωδικούς για να διαχωρίζει τις μεταδόσεις που πραγματοποιούνται στην ίδια συχνότητα. Το IS-95 είναι το μόνο 2G CDMA πρότυπο που έχει χρησιμοποιηθεί εμπορικά. Έχει χρησιμοποιηθεί σε χώρες όπως ΗΠΑ, Ιαπωνία, Νότια Κορέα, Σινγκαπούρη και άλλες Ασιατικές χώρες.

Το Personal Digital Cellular (PDC) είναι Ιαπωνικό 2G πρότυπο. Αρχικά ήταν γνωστό ως Japanese Digital Cellular (JDC), αλλά το όνομα άλλαξε σε PDC για να γίνει το σύστημα πιο ελκυστικό και εκτός Ιαπωνίας. Παρόλα αυτά το σύστημα χρησιμοποιήθηκε εμπορικά μόνο στην Ιαπωνία. Το specification είναι γνωστό ως RCR STD-27, και το σύστημα λειτουργεί σε δύο εύρη συχνοτήτων: 800 MHz και 1500 MHz.
Η έλλειψη επιτυχίας του PDC εκτός Ιαπωνίας έπαιξε σημαντικό ρόλο στην απόφαση των μεγάλων Ιαπωνικών εταιριών κατασκευής τηλεπικοινωνιακού εξοπλισμού να πετύχουν σε παγκόσμιο επίπεδο με το 3G, καθώς έχουν καινοτομήσει σε πολλούς τομείς που αφορά την ανάπτυξη του 3G. Το PDC ήταν πολύ δημοφιλές σύστημα στην Ιαπωνία και αυτή η επιτυχία ήταν ένας από του λόγους που οδήγησε τους Ιάπωνες στο να επιθυμούν να αναπτύξουν τα 3G συστήματα το συντομότερο δυνατό, καθώς η χωρητικότητα των PDC συστημάτων εξαντλούνταν [1].

1.2.1 Γενιά 2.5

Μεταξύ δεύτερης και τρίτης γενιάς (βλ. επόμενη υποενότητα) μπορούμε να διακρίνουμε την γενιά 2.5. Πρόκειται για την γενιά που περιέχει αναβαθμίσεις για τα 2G δίκτυα.

Ένα σύστημα GSM 2.5G περιλαμβάνει τουλάχιστον μία από τις τεχνολογίες High-Speed Circuit-Switched Data (HSCSD), General Packet Radio Services (GPRS) και Enhanced Data Rates for Global Evolution (EDGE).

Το μεγαλύτερο πρόβλημα στο GSM είναι ο χαμηλός ρυθμός μετάδοσης δεδομένων. Το βασικό GSM μπορούσε να παρέχει μόνο ρυθμό μετάδοσης στα 9.6-Kbps. Αργότερα καθορίστηκε ρυθμός στα 14.4-

Kbps, αλλά δεν χρησιμοποιήθηκε ευρέως. Όπως είναι φυσικό η περιήγηση στο ιντερνέτ με αυτούς τους ρυθμού είναι σχεδόν αδύνατη. Το HSCSD είναι ο ευκολότερος τρόπος για να επιταχύνουμε τα πράγματα. Αυτό σημαίνει ότι αντί για μια χρονοθυρίδα, ο κινητός σταθμός μπορεί να χρησιμοποιεί πολλές χρονοθυρίδες. Η μία χρονοθυρίδα μπορεί να παρέχει ρυθμό μετάδοσης δεδομένων είτε στα 9.6-Kbps είτε στα 14.4-Kbps. Στην εμπορική εφαρμογή του HSCSD μπορούν να χρησιμοποιηθούν μέχρι τέσσερις χρονοθυρίδες. Ο συνολικός ρυθμός μετάδοσης είναι ο αριθμός των χρονοθυρίδων επί τον ρυθμό μετάδοσης μιας χρονοθυρίδας. Το HSCSD είναι μία χαμηλού κόστους μέθοδος να αναβαθμίσουμε το GSM δίκτυο, καθώς απαιτεί αναβάθμιση μόνο του λογισμικού του δικτύου. Ένα μειονέκτημα που παρουσιάζει είναι ότι χρησιμοποιεί πολύτιμους πόρους του δικτύου. Το HSCSD είναι σύστημα μεταγωγής κυκλώματος και γι' αυτό δεσμεύει χρονοθυρίδες ακόμα και όταν κανένα δεδομένο δεν μεταδίδεται. Αυτό μπορεί να οδηγήσει σε ικανοποιητική μεταφορά δεδομένων πραγματικού χρόνου, αλλά αυξάνει την συμφόρηση στο δίκτυο. Ένα ακόμα αρνητικό στοιχείο είναι ότι οι κατασκευαστές κινητών συσκευών δεν έδειξαν ιδιαίτερο ενδιαφέρον στη δημιουργία συσκευών που υποστηρίζουν HSCSD.

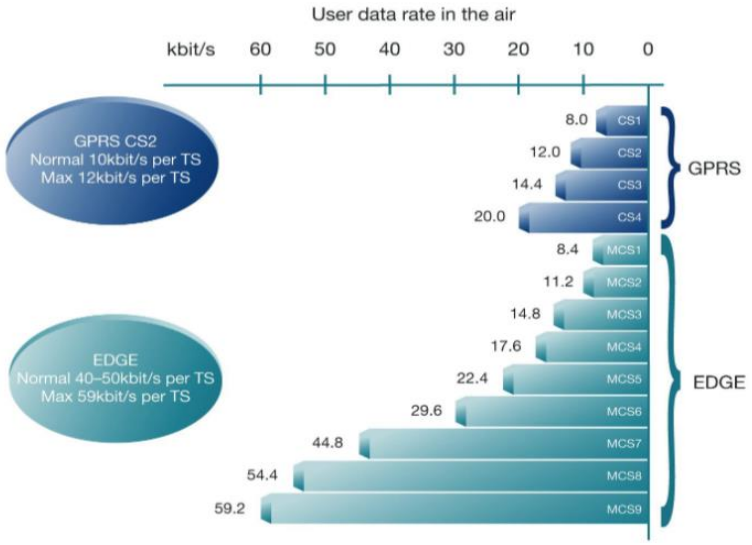
Η επόμενη τεχνολογία βελτίωσης της 2G υποδομής δικτύου, είναι το GPRS. Σε αυτήν την τεχνολογία ο, θεωρητικά και κάτω από ιδανικές συνθήκες, μέγιστος ρυθμός μετάδοσης δεδομένων μπορεί να φτάσει στα 115 - Kbps. Μία καλή προσέγγιση του throughput σε μέσες συνθήκες είναι 10 - Kbps ανά χρονοθυρίδα. Το GPRS είναι μέθοδος που χρησιμοποιεί μεταγωγή πακέτων και δεν δεσμεύει συνεχώς πόρους του δικτύου, αλλά μόνο όταν χρειάζεται κάτι να στείλει. Επίσης, είναι ιδιαίτερα αποτελεσματικό για εφαρμογές μη πραγματικού χρόνου, όπως εφαρμογές e-mail και το σερφάρισα στο ιντερνέτ.
Η εφαρμογή του GPRS είναι έχει υψηλότερο κόστος από την εφαρμογή του HSCSD, καθώς προϋποθέτει την προσθήκη νέων στοιχείων στην υποδομή του δικτύου 2G αλλά και την τροποποίηση των υπαρχόντων.

.

Τέλος, το EDGE αποτελεί μία ακόμα τεχνολογία βελτίωσης της 2G υποδομής δικτύου. Βάση του EDGE είναι η μέθοδος διαμόρφωσης 8-PSK (8-Phase Shift Keying). Με τη χρήσης του EDGE ένα GSM σύστημα αυξάνει τρείς φορές τον ρυθμό μετάδοσης δεδομένων. Το EDGE είναι ένας ελκυστικός τρόπος αναβάθμισης του GSM, καθώς απαιτεί μόνο την αναβάθμιση του λογισμικού των σταθμών βάσης, δεν έχει κάποια επιρροή στις υπάρχουσες κυψέλες και δεν απαιτεί νέα φάσματα συχνοτήτων. Επίσης, επειδή συνυπάρχει με την μέθοδο διαμόρφωσης Gaussian Minimum Shift Keying (GMSK) , οι χρήστες κινητών τηλεφώνων δεν είναι απαραίτητο να αλλάξουν τις συσκευές τους εάν δεν χρειάζονται τις υπηρεσίες που παρέχει το EDGE.

Όταν το EDGE χρησιμοποιείται μαζί με το GPRS, τότε ο συνδυασμός τους είναι γνωστός ως Enhanced GPRS (EGPRS). Ο μέγιστος ρυθμός μετάδοσης του EGPRS είναι 384 - Kbps κάτω από ιδανικές συνθήκες.

Το EDGE μεταδίδει δεδομένα με υψηλότερο bit-rate ανά κανάλι, όπως φαίνεται και στην Εικόνα 1.1 [4].

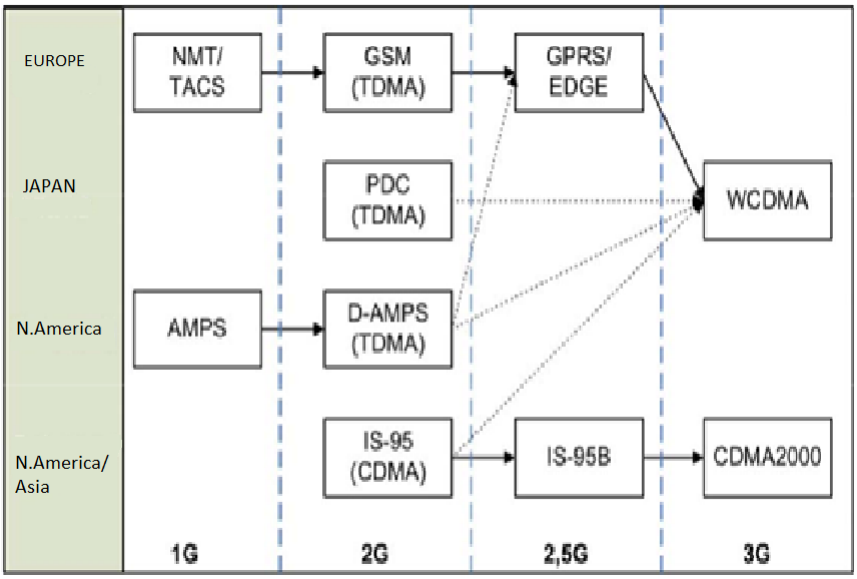


Εικόνα 1. 1: Το EDGE είναι τέσσερις φορές πιο αποτελεσματικό απ' ότι το GPRS. Τα bitrate που φαίνονται είναι ανά χρονοθυρίδα, σήμερα (2007) τα τερματικα μπορούν να λαμβάνουν δεδομένα από μέγιστα πέντε χρονοθυρίδες [4].

Ο συνδυασμός των τριών μεθόδων μπορεί να δημιουργήσει ένα ισχυρό σύστημα που μπορεί να συγκριθεί με τις πρ ώιμες μορφές εφαρμογής του συστήματος τρίτης γενιάς.

1.3 Γενιά 3

Η τρίτη γενιά εξέλιξης των τηλεπικοινωνιακών δικτύων βασίζεται στο GSM για δύο κύριους λόγους: το GSM έχει κυριαρχήσει στην αγορά και οι μεγάλες επενδύσεις που έχουν πραγματοποιηθεί στο GSM πρέπει να χρησιμοποιηθούν όσο το δυνατόν περισσότερο [5]. Στην Εικόνα 1.2 μπορούμε να δούμε την εξέλιξη των προτύπων των κινητών κυψελωδών δικτύων μέχρι την τρίτη γενιά [6].



Εικόνα 1.2: Η εξέλιξη των τηλεπικοινωνιακών προτύπων μέχρι το 3G,[5]

Τα δίκτυα τρίτης γενιάς δίνουν την δυνατότητα στον χρήστη να μετακινείται και να συνεχίζει να εξυπηρετείται ακόμα και από περιοχές που δεν υπάρχει κάλυψη από συστήματα τρίτης γενιάς αλλά υπάρχουν άλλου τύπου ασύρματα δίκτυτα , όπως κυψελωτά δίκτυα άλλης γενιάς, ασύρματα οικιακά δίκτυα, ή δορυφορικά δίκτυα.

Ο χρήστης δικτύων τρίτης γενιάς μπορεί να χρησιμοποιεί υπηρεσίες διαδικτύου, πολυμεσικές εφαρμογές με υψηλούς ρυθμούς μετάδοσης δεδομένων που ξεκινούν από τα 144Kbps και μπορεί να φτάνουν μέχρι μερικά Mbps.

1.3.1 Αρχιτεκτονική συστήματος τρίτης Γενιάς

Η κύρια ιδέα πίσω από το 3G είναι η δημιουργία μίας καθολικής υποδομής, ικανής να εξυπηρετεί τόσο τις υπάρχουσες όσο και τις μελλοντικές υπηρεσίες. Ο διαχωρισμός της τεχνολογίας πρόσβασης, της τεχνολογίας μεταφοράς, της τεχνολογίας υπηρεσιών και των εφαρμογών χρήστη μπορεί να οδηγήσει στην ικανοποίηση αυτού του αιτήματος.

Σε ένα δίκτυο τρίτης γενιάς μπορεί να εφαρμόζεται τόσο μεταγωτή πακέτου όσο και μεταγωγή κυκλώματος. Από την πλευρά της αρχιτεκτονικής πρωτοκόλλων, το 3G δίκτυο μπορεί να χωριστεί σε δύο στρώματα: το στρώμα πρόσβασης (access stratum) και το μη-προσβάσιμο στρώμα (non-access stratum). Το στρώμα πρόσβασης περιέχει πρωτόκολλα που χειρίζονται λειτουργίες του εξοπλισμού του χρήστη (User Equipment - UE) και του δικτύου πρόσβασης. Τα πρωτόκολλα του non-access stratum χειρίζονται λειτουργίες μεταξύ του UE και του δικτύου κορμού (core network).

Στο 3G, ο UE αποτελείται από δύο ξεχωριστά κομμάτια, τον κινητό εξοπλισμό και τη UMTS Service Identity Module (USIM) κάρτα. Το υποσύστημα που ελέγχει την ευρείας ζώνης (wideband) ασύρματη πρόσβαση έχει διαφορετικά ονόματα ανάλογα με την ασύρματη τεχνολογία που χρησιμοποιείται. Όταν αναφερόμαστε σε UMTS με WCDMA ασύρματη πρόσβαση, το όνομα που χρησιμοποιούμαι είναι Universal Terestrial Radio Access Network (UTRAN). Ο άλλος τύπος ασύρματης πρόσβασης δικτύου που περιλαμβάνεται στο UMTS λέγεται GSM EDGE Radio Access Network (GERAN). Ο καθορισμός των προδιαγραφών του GERAN και ο εναρμονισμός του με το UTRAN πραγματοποιείτα στα 3GPP R4 και 3GPP R5.

Το UTRAN απαρτίζεται από υποσυστήματα δικτύου ραδιοκάλυψης (Radio Network Subsystems - RNS)Ενα RNS αποτελείται από ένα σύνολο από radio στοιχεία καθώς και από το αντίστοιχο στοιχείο ελέγχου τους. Στο UTRAN το radio στοιχείο είναι το Node B, αναφέρεται ως Base Station (BS), και το στοιχείο ελέγχου είναι το Radio Network Controller (RNC). Τα RNS συνδέονται μεταξύ τους μέσω της διεπαφή Iur [5].

Το GERAN χωρίζεται σε ένα ή περισσότερα BS με τις απαραίτητες διεπαφές για να επικοινωνεί με το περιβάλλον του (MSs,core network, άλλα GERAN ή UTRAN). Ένα GERAN BS αποτελείται από ένα Base Transceiver Station (BTS) και από ένα Base Station Controller (BSC). Επειδή ο τρόπος με τον οποίο θα χρησιμοποιείται το BS δεν καθορίζεται από κάποιο specification, η εσωτερική αρχιτεκτονική του καθορίζεται από τον εκάστοτε κατασκευαστή. Οι εξωτερικές διεπαφές του BS πρέπει να επιτρέπουν την συνεργασία μεταξύ εξοπλισμού που παρέχεται από διαφορετικούς κατασκευαστές. Σε μία διεπαφή η επικοινωνία μεταξύ δύο στοιχείων του δικτύου διασφαλίζεται από πρωτόκολλα που ακολουθούν καθορισμένη αρχιτεκτονική [8].

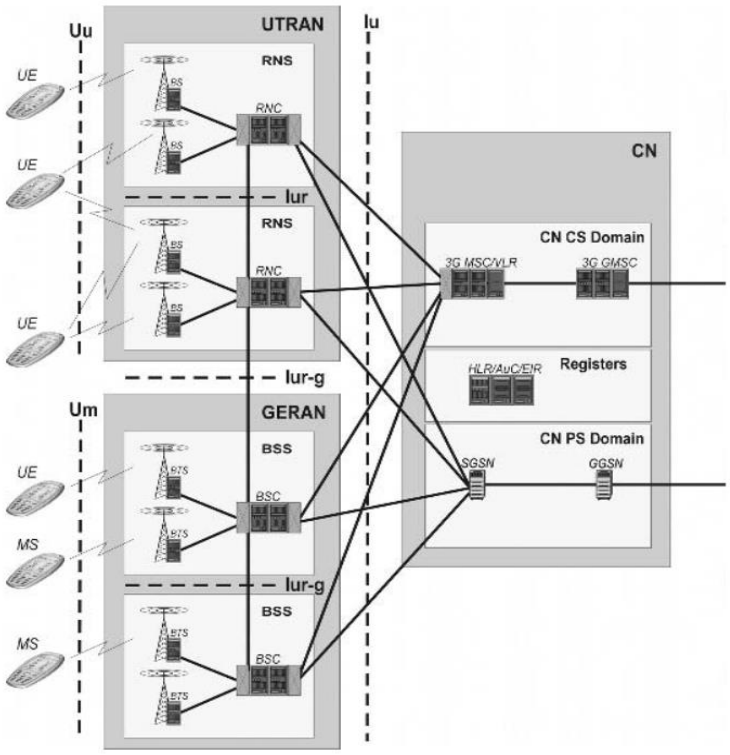
Βασική αρχή του GERAN είναι ο διαχωρισμός των λειτουργίων του Δικτύου Πρόσβασης (Radio Access Network - RAN) από τις λειτουργίες του Δικτύου Κορμού (Core Network -CN). Βασική αρχή του GERAN είναι ο διαχωρισμός των λειτουργίων του Δικτύου Πρόσβασης (Radio Access Network - RAN) από τις λειτουργίες του Δικτύου Κορμού (Core Network -CN Επιπλέον, η συνήοαρξη του GERAN με το UTRAN, προύποθέτει ότι το GERAN πρέπει να παρέχει το ίδιο σύνολο υπηρεσιών με το UTRAN. Παράλληλα, το GERAN πρέπει να υποστηρίζει ένα περιβάλλον αποτελούμενο από πολλαπλούς κατασκευαστές [7].

Σημειώνουμε ότι, με το όρο Δίκτυο Κορμού καλύπτουμε όλα τα στοιχεία του δικτύου που χρειάζονται για τις λειτουργίες της μεταγωγής και του ελέγχου του συνδρομητή. Στις αρχές του UMTS τα στοιχεία αυτά κληρονομήθηκαν από το GSM και τροποποιήθηκαν για το UMTS. Αργότερα όταν η τεχνολογία μεταφοράς άλλαξε, άλλαξε και η εσωτερική δομή του δικτύου κορμού.

Τα στοιχεία του circuit switch (CS) domain του δικτύου κορμού είναι ικανά να χειρίζονται τόσο 2G όσο 3G συνδρομητές. Αυτό απαιτεί αλλαγές στα MSC/VLR και HLR/AC/EIR. Για παράδειγμα, οι μηχανισμοί ασφαλείας στήνονται διαφορετικά στο 2G και στο 3G και τα στοιχεία του circuit switch domain πρέπει να είναι ικανά να διαχειρίζονται και τις δύο περιπτώσεις.

Το packet switch (PS) domain είναι να εξελιγμένο GPRS σύστημα. Αν και τα ονόματα των στοιχείων του δικτύου είναι ίδια με αυτά του 2G, η λειτουργία τους δεν είναι η ίδια. Η σημαντικότερη διαφορά είναι στο SGSN. Στο 2G η κύρια λειτουργία του SGSN είναι η διαχείρηση κινητικότητας (mobility management), ενώ στο 3G η λειτουργία αυτή χωρίζεται μεταξύ των στοιχείων του RNC και του SGSN. Αυτό σημαίνει ότι κάθε αλλαγή κυψέλης που κάνει ο συνδρομητής στο UTRAN δεν είναι απαραίτητα ορατή στο packet switch domain, αλλά ο RNC χειρίζεται την κατάσταση.

Στην Εικόνα 1.3 μπορούμε να δούμε την αρχιτεκτονική του δικτύου UMTS καθώς και τις διεπαφές μεταξύ των στοιχείων του δικτύου[5]. GERAN και UTRAN ενώνονται μέσω της διεπαφής Iur-g.



Εικόνα 1. 3: Αρχιτεκτονική Δικτύου UMTS-στοιχεία του δικτύου και οι συνδέσεις τους για μεταφορά δεδομένων χρήστη, [5]

Εκτός από το UTRAN που βασίζεται στο WCDMA και το GERAN, υπάρχει και η τεχνολογία CDMA2000 που χρησιμοποιεί εύρος ζώνης ίσο με 1.25MHz, το οποίο αυξάνει τις υπηρεσίες φωνής και δεδομένων, πλοήγησης στο Internet αλλά και τις υπηρεσίες πολυμέσων. Αυτή η τεχνολογία διπλασιάζει την χωρητικότητα του χρήστη από την τεχνολογία cdmaOne. Σαν εξέλιξη του CDMA2000 το 3GPP2 πρώτα εισήγαγε την High Rate Packet Data(HRPD) η οποία αναφέρεται ως CDMA20001xEV-DO. Αυτό το πρότυπο επιτρέπει υψηλές ταχύτητες, τεχνικές μεταγωγής πακέτων και μέγιστο ρυθμό δεδομένων πάνω από 2Mbps.

Το 3GPP ακολούθησε παρόμοια κατεύθυνση εισάγοντας βελτιώσεις στο σύστημα WCDMA παρέχοντας το High-Speed Downlink Packet Access (HSDPA) το οποίο έφερε μεγαλύτερη αποδοτικότητα φάσματος για υψηλότερης ταχύτητας υπηρεσίες δεδομένων. Το πρότυπο High-Speed UplinkPacket Access (HSUPA) προτάθηκε αργότερα και ο συνδυασμός των HSDPA και HSUPA ονομαστικέ HSPA[9][2].

Η τελευταία εξέλιξη του HSPA είναι το HSPA+ το οποίο καθορίστηκε ως αποτέλεσμα της πρόσθεσης Multiple Input/Multiple Output (MIMO) δυνατοτήτων κεραίας και 16 QAM (uplink)/64 QAM (downlink) διαμόρφωσης. Ζευγαρωμένο με βελτιώσεις στο δίκτυο ασύρματης πρόσβασης , το HSPA+ επιτρέπει ταχύτητες uplink στα 11 Mbps και downlink στα 42 Mbps[2].

1.4 Γενιά 4

Η τεχνολογία 4G [2] επιτρέπει να έχουμε μεγαλύτερη ταχύτητα μεταφοράς δεδομένων τόσο στο uplink όσο και στο downlink, καθώς και αύξηση του τύπου του περιεχομένου που είναι διαθέσιμο για τους χρήστες των κινητών συσκευών. Τα 4G δίκτυα είναι λύση βασισμένη στο IP για την εξηπηρέτηση υπηρεσιών φωνής, δεδομένων, και πολυμέσων σε χρήστες κνητών συσκευών. Παρέχουν βελτιομένους ρυθμούς δεδομένων σε σχέση με προηγούμενες γενιές ασύρματης τεχνολογίας. Ταχύτερες ασύρματες συνδέσεις ευρείας ζώνης επιτρέπουν υψηλότερες υπηρεσίες δεδομένων, όπως επιχειρηματικές εφαρμογές, video και ήχο, video μηνύματα, mobile TV και παιχνίδια.

Το 3GPP άρχισε την έρευνα για την προτυποποίηση του Long Term Evolution (LTE) από το 2004.Τα πλεονεκτήματα του LTE περιγράφονται παρακάτω:

1. Υψηλή αποτελεσματικότητα φάσματος (spectral efficiency).
2. Χαμηλός χρόνος καθυστέρησης (latency).
3. Υποστήριξη μεταβλητού εύρους ζώνης.
4. Απλή αρχιτεκτονική πρωτοκόλλων.
5. Συμβατότητα και συνεργασία με προηγούμενες εκδόσεις του 3GPP.
6. Διεργασία με άλλα συστήματα, όπως το cdma2000.
7. Μεθόδους πολυπλεξίας FDD και TDD.
8. Αποτελεσματική μετάδοση δεδομένων με multicast/broadcast μεθόδους.

1.4.1 LTE και άλλες ευρυζωνικές ασύρματες τεχνολογίες

Πραγματικός στόχος του LTE είναι να έχει μέγιστη ρυθμαπόδοση (throughput) στα 100Mbps uplink και 50 Mbps στο downlink.Το 2008 τα specification στο 3GPP release 8 ήταν αρκετά σταθερά για το πρώτο κύμα εξοπλισμού LTE. Το 2009 οι partners του 3GPP έκαναν επίσημη αίτηση στην ITU προτείνοντας το LTE release 10 and beyond(LTE-Advanced). Οι βελτιώσεις που παρέχει το LTE-Advanced μπορούν να θεωρηθούν οι παρακάτω:

- Παγκόσμιες λειτουργίες και υπηρεσίες roaming
- Συμβατότητα υπηρεσιών
- Συνεργασία με άλλα συστήματα ασύρματης πρόσβασης
- Βελτιωμένοι ρυθμοί μετάδοσης δεδομένων που φτάνουν στα 100 Mbps για υψηλή κινητικότητα και 1 Gbps για χαμηλή κινητικότητα.

Το LTE δεν είναι η μόνη λύση για την παροχή ευρυζωνικών ασύρματων κινητών τεχνολογιών. Αρκετές λύσεις, ιδιαίτερα για σταθερρές εφαρμογές, υπάρχουν στην αγορά. Υπάρχουν και τυποποιημένες

εναλλακτικές λύσεις που μερικώς επικαλύπτονται με το LTE.Τα πιο γνωστά τέτοια συστήματα είναι τα IEEE 802.11 WiFi συστήματα.

- WiMAX

Η IEEE ανέπτυξε το πρότυπο IEEE 802.16 ή Worldwide Interoperability for Microwave Access (WiMAX) με σκοπό την παγκόσμια εφαρμογή των Wireless Metropolitan Area Networks. Το WiMAX είναι διαθέσιμο σε δύο εκδόσεις – σταθερή και κινητή. Το σταθερό WiMAX το οποίο βασίζεται στο πρότυπο IEEE 802.16-2004 είναι ιδανικό για την μεταφορά ασύρματων σταθερών ευρυζωνικών υπηρεσιών, είναι παρόμοιο του DSL ή των υπηρεσιών που παρέχει το ενσύρματο modem. Το κινητή WiMAX το οποίο βασίζεται στο IEEE 802.16e standard, υποστηρίζει τόσο σταθερές όσο και κινητές εφαρμογές ενώ προσφέρει στους χρήστες βελτιωμένη απόδοση, χωρητικότητα και κινητικότητα.

- WiFi

Το Wireless Fidelity (WiFi) σύστημα χρησιμοποιείται για να παρέχει ασύρματες ευρυζωνικές υπηρεσίες στο χρήστη. Βασίζεται στην οικογένεια τυποποιήσεων IEEE 802.11 και ο πρωταρχικός του ρόλος είναι Τοπικό Δίκτυο(LAN) σχεδιασμένο να παρέχει ευρυζωνική κάλυψη στα όρια ενός κτιρίου. Το WiFi παρέχει υψηλότερο ρυθμό δεδομένων απ' ότι τα συστήματα 3G, αλλά το WiFi δεν είναι σχεδιασμένο να υποστηρίζει την κινητικότητα και έχει μειωμένη χωρητικότητα λόγω του ανεπαρκούς πρωτοκόλλου Carrier Sense Multiple Access (CSMA) που χρησιμοποιεί.

Το κυριότερο πλεονέκτημα του WiFi είναι η μεγάλη διαθεσιμότητα τερματικών συσκευών. Διεπαφές WiFi έχουν αρκετές συσκευές όπως laptop, smart phones, κάμερες, φωτογραφικές μηχανές κ.α.

1.4.2 Τα πιο σημαντικά χαρακτηριστικά του LTE

Τα πιο σημαντικά χαρακτηριστηκά της τεχνολογίας LTE είναι τα ακόλουθα:

- Χρήση της μεθόδου πολυπλεξίας καναλιών OFDM στο φυσικό επίπεδο για το downlink.Χρήση τεχνολογίας διαμόρφωσης SC-FDMA στο uplink, για να έχουμε χαμηλό Peak-to-Average Power Ratio (PAPR).
- Υποστήριξη τεχνολογιών TDD (Time Division Duplexing) και FDD (Frequency Division Duplexing). Το TDD επιλέγεται από αρκετές εφαρμογές λόγω της ευελιξίας στην επιλογή της αναλογίας ρυθμού μετάδοσης μεταξύ downlink και uplink, την ικανότητα εκμετάλλευσης του channel reciprocity[1], λιγότερο πολύπλοκος σχεδιασμός πομποδεκτών.
- Υποστήριξη AMC (Adaptive Modulation and Coding). Το AMC είναι ένας αποτελεσματικός μηχανισμός για να αυξήσουμε τη ρυθμαπόδοση σε ένα κανάλι μεταβλητού χρόνου. Ο μηχανισμός δίνει την δυνατότητα να παρέχεται στο χρήστη με τον υψηλότερο δυνατό ρυθμό δεδομένων που μπορεί να υποστηρίξει η ζεύξη του.
- Υποστήριξη μεταβλητού εύρους ζώνης. Το E-UTRA πρέπει να λειτουργεί σε τοποθετήσεις φάσματος διαφορετικών μεγεθών, συμπεριλαμβανομένου 1.25, 1.6, 2.5, 5, 10, 15, και 20 MHz τόσο στο uplink όσο και στο downlink. Η προσαρμογή μπορεί να γίνει δυναμικά για να υποστηρίξει την μετάβαση του χρήστη μεταξύ διαφορετικών δικτύων που μπορεί να έχουν διαφορετικά εύρη συχνοτήτων.

[1] "Channel reciprocity is an inherit feature of time division duplex (TDD) system, which is widely used to get uplink (UL)/downlink (DL) channel knowledge from DL/UL channel measurements without additional feedback", Y Han, J Ni, GK Du - Communications and Networking in China (CHINACOM), 2010 5th International ICST Conference, 2010

- Πολύ υψηλό ρυθμό μετάδοσης. Το LTE είναι ικανό να υποστηρίξει ρυθμούς μετάδοσης της τάξης των 100Mbps στο downlink σε φάσμα των 20MHz και 50Mbps στο uplink στα 20MHz φάσμα.
- Κινητικότητα. Το E-UTRAN πρέπει να παρέχει βέλτιστη απόδοση για χαμηλή ταχύτητα κινητικότιτας, της τάξης 0-15 km/h.Υψηλότερες ταχύτητες κινητικότητας-15 έως 120 km/h- θα πρέπει να υποστηρίζονται. Η κινητικότητα θα πρέπει να διατηρείται και σε ταχύτητες από 120 έως 350 km/h.
- Επαναμεταδόσεις επιπέδου ζεύξης. Το LTE υποστηρίζει Automatic Retransmission Requests (ARQ) στο επίπεδο ζεύξης. Κάθε πακέτο που μεταδίδεται πρέπει να επιβεβαιώνεται από τον δέκτη. Τα ανεπιβεβαίωτα πακέτα θεωρούνται χαμένα και ξαναστέλνονται.
- Ταυτόχρονη υποστήριξη χρηστών. Το LTE δίνει την δυνατότητα υποστήριξης πολλαπλών χρηστών σε μια χρονοθυρίδα.
- Ασφάλεια. Η ασφάλεια που παρέχει το LTE είναι βελτιωμένη σε σύγκριση με τα προηγούμενα συστήματα. Η χρήση των UICC SubscriberIdentityModule(SIM) και η χρησιμοποίηση ιδιωτικών κλειδιών για πιστοποίηση κάνει το LTE πιο ασφαλές σύστημα από συστήματα προηγουμένων γενιών.
- Αποτελεσματικές παγκόσμιες υπηρεσίες roaming. Επειδή το LTE είναι ένα ενοποιημένο πρότυπο 4G για τους περισσότερους παρόχους, οι συσκευές του LTE είναι πιο εύκολο να σετάρονται για υπηρεσίες roaming.

1.5 Γενιά 5η

Το 5G είναι εξέλιξη του 4G. Σκοπός του είναι να γίνει αποτελεσματικότερο και ταχύτερό από το 4G, να υποστηρίζει περισσότερους χρήστες, περισσότερες συσκευές, περισσότερες υπηρεσίες και να παράγει χαμηλότερες εκπομπές διοξειδίου του άνθρακα.

Η επικοινωνία μεταλλάσσεται από επικοινωνία από άνθρωπο προς άνθρωπο σε οτιδήποτε προς οτιδήποτε.

Μια από τις μεγαλύτερες αλλαγές στα 5G δίκτυα θα είναι η επικοινωνία μεταξύ μηχανών. Τα πάντα θα είναι συνδεδεμένα, κράνη ποδηλάτων, συστήματα ύδρευσης, σοδειές, οικονομικές δομές και είδη προς εξαφάνιση.

Η ενέργειες προτυποποίησης αναμένεται να ξεκινήσει το 2016,οδηγόντας σε εμπορική διαθεσιμότητα του εξοπλισμού και των συσκευών έως το 2020. Το 5G δεν είναι μια τεχνολογία που θα απαιτήσει ριζικές αναβαθμίσεις εξοπλισμού. Το κόστος της υποδομής για την ανάπτυξη και χρήση της τεχνολογίας αυτής αναμένεται να κυμανθεί περίπου στα 1.5 δισεκατομμύρια δολάρια για τη χώρα της Νότιας Κορέας[27]. Ανάλογο αναμένεται το κόστος σε αντίστοιχες πληθυσμιακά χώρες.[7] Στόχος του 5G είναι να χτίσει πάνω στα υπάρχοντα τηλεπικοινωνιακά συστήματα. Θα φέρει εξελιγμένες εκδόσεις της υπάρχουσας ασύρματης τεχνολογίας, το συνδυασμό των τεχνολογιών cloud και core μαζί με συμπληρωματικές τεχνολογίες για την διαχείριση περισσότερης κίνησης, περισσότερων συσκευών διαφόρων τύπων, με διάφορες λειτουργικές απαιτήσεις.

Οι παράμετροι πάνω στις οποίες η τεχνολογία 5G θα αναπτυχθεί περιλαμβάνουν:

- Την χωρητικότητα του συστήματος.
- Ρυθμαπόδοση δεδομένων.
- Αξιοπιστία δεδομένων.
- Καθυστέρηση.
- Κατανάλωση ενέργειας.
- Σύγκλιση τεχνολογιών.
- Έξυπνη επικοινωνία .

Το 5G, όπως και κάθε προηγούμενη γενιά κινητών τηλεπικοινωνιών, για να βελτιώσει την εμπειρία του χρήστη θα χρησιμοποιεί επιπλέον ζώνες συχνοτήτων. Ο έλεγχος του φάσματος συχνοτήτων μέσω

αδειοδοτήσεων είναι σημαντικό για να αποφεύγουμε τις παρεμβολές. Επίσης παρέχει ένα βασικό επίπεδο ασφάλειας και επιτρέπει στους παρόχους να φτάσουν τις απαιτήσεις της ποιότητας υπηρεσιών. Η χρήση των περισσότερων συχνοτήτων κάτω από τα 6GHz θα ικανοποιήσουν μόνο μέρος των αναγκών και μπάντες συχνοτήτων από τα 10GHz και πάνω θα χρησιμοποιηθούν.

Παγκόσμιος εναρμονισμός της νομοθεσίας αδειοδότησης και του αντίτιμου για τις μπάντες συχνοτήτων θα ήταν ευνοϊκό για τους τηλεπικοινωνιακούς παρόχους.

Ένας από τους κύριους λόγους της πολυπλοκότητας των δικτύων βασίζεται στον παραδοσιακό τρόπο με τον οποίο έχει αναπτυχθεί η τεχνολογία των δικτύων. Ο σχεδιασμός στοιχείων του δικτύου όπως τα routers και τα switches είναι κλειστός, καθώς τείνουν να έχουν τα δικά τους συστήματα διαχείρισης με κάθετη ενσωμάτωση των στοιχείων ελέγχου και προώθησης (forwarding). Ο στόχος της διαχείρισης δικτύου, όμως, είναι να διασφαλίζει ότι ολόκληρο το δίκτυο συμπεριφέρεται όπως πρέπει-ένας αντικειμενικός στόχος που είναι αρκετά πιο σημαντικός από τις ιδιότητες του κάθε στοιχείου του δικτύου χωριστά.

Η προσέγγιση του Software-Defined Networking (SDN), παρέχει την δυνατότητα προγραμματισμού στο επίπεδο όλου του δικτύου που θα επιτρέπει στους παρόχους να διαχειρίζονται δίκτυα με ελεγχόμενο τρόπο και θα επιτρέπουν στα δίκτυα να προσαρμόζονται σε νέες εφαρμογές και χρήστες. Τεχνολογίες virtualization όπως η Network Functions Virtualization (NFV), παρέχουν virtualization και επίπεδα αφαίρεσης για το σύνολο των πόρων.

Σκοπός των τεχνολογιών SDN και NFV είναι να παρέχουν λειτουργίες, δίκτυα και υποδομές ως υπηρεσίες αντί ως ενσωματωμένα χαρακτηριστικά συστημάτων. Το χτίσιμο συστημάτων σε κουτιά δεν είναι καινούργια τεχνική, αλλά ο διαχωρισμός του επιπέδου ελέγχου και δεδομένων στα τηλεπικοινωνιακά δίκτυα είναι σημαντική αλλαγή στο σχεδιασμό των δικτύων που θα παρέχουν τη δυνατότητα στους παρόχους να χτίζουν ευέλικτα δίκτυα.

Όσον αφορά την αντιμετώπιση των σφαλμάτων τα 5G συστήματα θα αυτοθεραπεύονται μέσω εφαρμογών χωρίς να επηρεάζουν την αξιοπιστία του δικτύου.

Η Self-Organizing Networks (SON) ιδέα όπως είναι σήμερα, η οποία επιτρέπει στα base stations να διαμορφώνονται αυτόματα συμπεριλαμβανομένου και του Automatic Neighbor Relations (ANR), θα αναπτυχθούν πολύ περισσότερο στα 5G συστήματα. Προηγμένες SON τεχνικές δεν θα εφαρμόζονται μόνο στα φυσικά στοιχεία του δικτύου, θα επιτρέπουν στους παρόχους, για παράδειγμα, να εξισορροπούν το φορτίο σε περιβάλλοντα που πολλαπλές τεχνολογίες ασύρματης πρόσβασης είναι παρούσες, και θα υποστηρίζουν την κατεύθυνση της κίνησης καθώς και την δυναμική τοποθέτηση φάσματος.

Τα τριχωειδή δίκτυα (capillary networks) είναι δίκτυα ασύρματης τεχνολογίας μικρού εύρους που παρέχουν τοπική συνδεσιμότητα σε σύνολο συσκευών κοινών λειτουργιών ή κοινού σκοπού και συνδέονται στην παγκόσμια επικοινωνιακή υποδομή μέσω της τριχωειδούς πύλης. Το να συνδέσουμε δισεκατομμύρια συσκευές κατευθείαν σε ένα κυψελώδη σύστημα δεν είναι δυνατό, τα τριχωειδή–δίκτυα παρέχουν έναν αποτελεσματικό τρόπο να αποφορτίσουν διαδικασίες συνδεσιμότητας για ένα σύνολο συσκευών με κοινό σκοπό και κοινή γεωγραφική περιοχή.

Κεφάλαιο 2ο

2.1 Περιγραφή SDN

Ο μεγάλος όγκος μεταφοράς δεδομένων , η εξέλιξη των τηλεπικοινωνιακών δικτύων καθώς και η έκρηξη των υπηρεσιών cloud έχει δημιουργήσει την ανάγκη ανάπτυξης τεχνολογιών επανακαθορισμού των δομών των δικτύων.

Τα δίκτυα σήμερα είναι συνδυασμός δομών επιπέδου ελέγχου και δεδομένων, χωρίς να υπάρχει ξεκάθαρος διαχωρισμός μεταξύ τους. Στοιχεία του δικτύου, όπως τα switches, έχουν εξελιχθεί να υποστηρίζουν μια ποικιλία από πρωτόκολλα αυξάνοντας το κόστος χρήσης και ανάπτυξης τους. Επίσης, η έλλειψη υπολογιστικών πόρων οδήγησε στην δημιουργία διανεμημένων πρωτοκόλλων τα οποία δουλεύουν καλά σε περιορισμένα όρια, αλλά δεν καλύπτουν τις σημερινές ανάγκες της γιγάντωσης των δικτύων.

Τα switches και οι routers σήμερα περιέχουν λογισμικό το οποίο είναι διαφορετικό ανάλογα με τον κατασκευαστή του μηχανήματος. Το λογισμικό που υπάρχει στα σημερινά μηχανήματα του δικτύου δεν καθορίζει δύο σημαντικές παραμέτρους τις οποίες αποζητούν οι διαχειριστές των δικτύων.

- Την ικανότητα για γρήγορη αλλαγή του δικτύου ώστε να μπορεί να προσαρμόζεται στις ανάγκες χρήσης (υψηλότερες ταχύτητες, διαφορετικοί τύποι υπολογισμού των μονοπατιών κ.τ.λ.)
- Βελτιστοποίηση κόστους.

Στα σημερινά δίκτυα τα επίπεδα ελέγχου και δεδομένων συνδυάζονται σε ένα κόμβο του δικτύου. Το επίπεδο ελέγχου είναι υπεύθυνο για το configuration του κόμβου, αλλά και για τον υπολογισμό των μονοπατιών που θα χρησιμοποιηθούν από τις ροές δεδομένων. Η προώθηση των δεδομένων στο επίπεδο του υλικού βασίζεται σε αυτές τις πληροφορίες ελέγχου.

Με αυτήν τη προσέγγιση, από την στιγμή που καθοριστεί η πολιτική προώθησης, ο μόνος τρόπος για να κάνουμε ρυθμίσεις σε αυτήν την πολιτική είναι μέσω αλλαγών στο configuration της συσκευής.

Το SDN είναι μια τεχνολογία η οποία δεν επινοήθηκε τα τελευταία χρόνια, αλλά η έννοια της εξελίσσεται από το 1996 και μετά. Πιο πρόσφατα, το Ethane (2007) και το OpenFlow (2008) έφεραν την εφαρμογή του SDN πιο κοντά στην πραγματικότητα. Το Ethane είναι μια αρχιτεκτονική ασφάλειας που συνδυάζει flow-based switches με έναν κεντρικό controller, ο οποίος διαχειρίζεται την αποδοχή και δρομολόγηση των ροών (flows). Το OpenFlow είναι το πρωτόκολλο που επιτρέπει την εισαγωγή ροών στο Flow Table μέσω εξωτερικών συσκευών. Ο Open Networking Foundation (ONF) [10] παρουσιάζει το SDN ως εξής: Στην αρχιτεκτονική του SDN τα επίπεδα του ελέγχου και της μεταφοράς διαχωρίζονται, η ευφυΐα του δικτύου και η κατάσταση το καθορίζονται κεντρικά από τον controller και η υποδομή του δικτύου αφαιρείται (abstracted) από τις εφαρμογές.¨

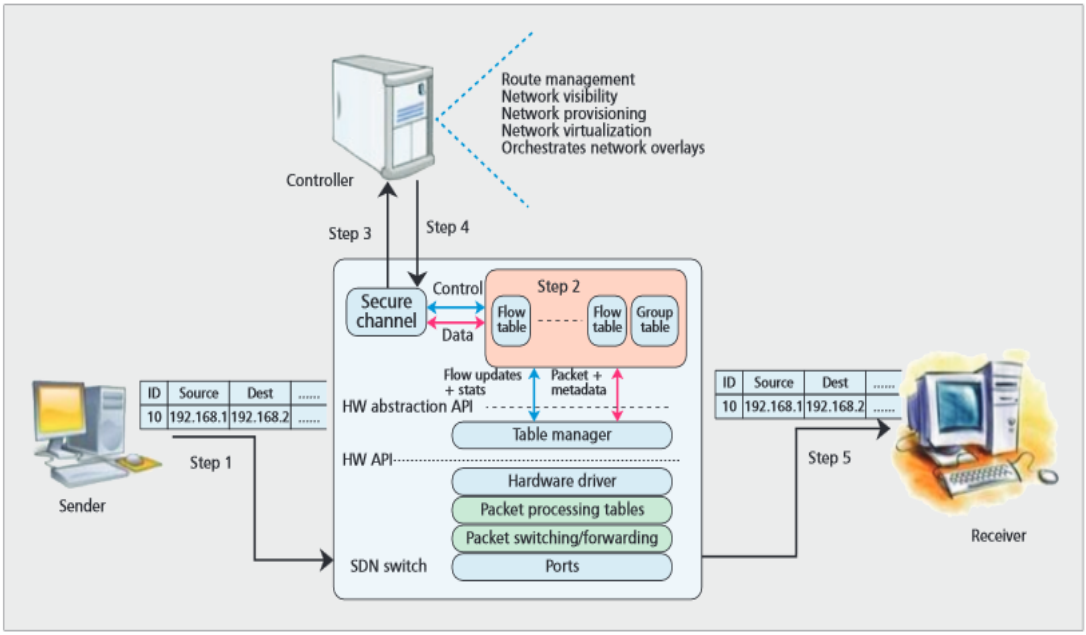
Το SDN βασίζεται σε τέσσερα κύρια χαρακτηριστικά.

- Τον διαχωρισμό του επιπέδου ελέγχου από το επίπεδο δεδομένων.
- Έναν κεντρικοποιημένο controller και μια συνολική οπτική του δικτύου.
- Open interfaces μεταξύ του επιπέδου ελέγχου (controllers) και των συσκευών του επιπέδου δεδομένων.
- Προγραμματισμό του δικτύου από εξωτερικές συσκευές.

2.2 Πως δουλεύει το SDN

Παρακάτω παρουσιάζουμε τον τρόπο λειτουργίας μεταξύ του controller και του switch στο SDN. Από την στιγμή που το πρώτο πακέτο φτάνει στην πόρτα του switch από τον αποστολλέα, το switch ελέγχει για κάποιον κανόνα ροής για αυτό το πακέτο. Εάν υπάρχει κάποιο ταίριασμα για αυτό το πακέτο στα flow tables του switch τότε εκτελείται η εντολή που αντιστοιχεί σε αυτήν την flow entry. Μετά το πακέτο πηγαίνει στον δέκτη.

Εάν δεν υπάρχει ταίριασμα στο flow table τότε το πακέτο μπορεί να προωθηθεί στον controller μέσω ασφαλούς καναλιού. Χρησιμοποιώντας την Southbound API (π.χ., OpenFlow) ο controller μπορεί να προσθέτει, ανανεώνει και να διαγράφει flow entries. Ο controller εκτελεί τον αλγόριθμο δρομολόγησης και προσθέτει νέες εισαγωγές ροών στο flow table στο switch αλλά και σε κάθε switch του flow path. Μετά το switch προωθεί το πακέτο στην κατάλληλη πόρτα και από εκεί στον δέκτη του πακέτου.



Εικόνα 2. 1: ***Η λειτουργία του SDN***

2.2.1 Προκλήσεις υποδομών που συναντά το SDN

Τρεις είναι οι κύριες προκλήσεις που αντιμετωπίζει η εφαρμογή του SDN. Η απόδοση (performance), η δυνατότητα προγραμματισμού (programmability) και η ευελιξία (flexibility).

Η απόδοση αναφέρεται στην ταχύτητα επεξεργασίας ενός κόμβου του δικτύου. Το programmability αναφέρεται στην ικανότητα να αλλάζει ή να δέχεται ένα σύνολο κανόνων με σκοπό την μεταβολή της συμπεριφοράς του δικτύου. Η ευελιξία αναφέρεται στην ικανότητα του συστήματος να προσαρμόζεται για να υποστηρίζει νέα χαρακτηριστικά, όπως νέα πρωτόκολλα, εφαρμογές.

Για να αντιμετωπιστούν αυτές οι προκλήσεις μπορούν να χρησιμοποιηθούν επεξεργαστές γενικού σκοπού (Central Prossesing Unit - CPU, General Processing Unit - GPU) οι οποίοι παρέχουν ευελιξία. Οι γλώσσες

προγραμματισμού υψηλού επιπέδου και διάφορα σχεδιαστικά εργαλεία παρέχουν το επίπεδο αφαίρεσης που χρειάζεται ώστε να αναπτύσσονται σύνθετες λειτουργίες επεξεργασίας πακέτων. Παρ' όλα αυτά οι CPUs υπολείπονται σε θέματα απόδοσης και κατανάλωσης ρεύματος.

Επίσης επεξεργαστές ροών δικτύου (Network Proccesing Units - NPU, Network Flow Processors - NFPs), που έχουν αναπτυχθεί με αρχιτεκτονικές για επεξεργασία δικτύου. Η ευελιξία είναι περιορισμένη καθώς μεγαλύτερη γνώση για την συσκευή είναι απαραίτητη για τον καθορισμό των λειτουργιών επεξεργασίας των πακέτων/ροών και πλήρης εκμετάλλευσης των δυνατοτήτων παράλληλης επεξεργασίας της συσκευής.

Οι προγραμματιζόμενες λογικές συσκευές (Programmable Logical Devices - PLD) διαμορφώνονται χρησιμοποιώντας σχεδιαστικά εργαλεία για το υλικό. Αυτή η τεχνολογία είναι ιδανική για την δημιουργία παράλληλων μονοπατιών δεδομένων που είναι προσαρμοσμένα για λειτουργίες επεξεργασίας μεμονωμένων δικτύων. Μπορούν να πετύχουν ταχύτητες επεξεργασίας μέχρι και 200 Gb/s ανά συσκευή.

Ολοκληρωμένα κυκλώματα σχετιζόμενα με εφαρμογές (Application Specific Integrated Circuits - ASIC)είναι ιδιόκτητες συσκευές που φτιάχνονται από κατασκευαστές όπως η Cisco και η Huawei, όταν τα υπάρχοντα προϊόντα και συσκευές δεν μπορούν να πετύχουν τους περιορισμούς απόδοσης που έχουν τεθεί. Τα ASICs παρέχουν χαμηλή ευελιξία ενώ παρέχουν υψηλή απόδοση.

Το να φτιάχνουμε πλατφόρμες βασιζόμενοι σε custom-built συσκευές (π.χ PLD ή Application Specific Standard Products - ASSP)σε συνδυασμό με NPU/NFP και CPU/GPU παρουσιάζει μια υβριδική αρχιτεκτονική. Μια τέτοια πλατφόρμα μπορεί να παρέχει γρήγορη προώθηση ροών μαζί με τον προγραμματισμό(programmability και την ελεγχόμενη επεξεργασία της κίνησης και των νέων ροών.

Ένας στόχος του SDN είναι η δημιουργία δικτύων που βασίζεται σε υλικό γενικού-σκοπού. Ο συνδυασμός τεχνολογιών όπως αναφέρθηκε στην υβριδική αρχιτεκτονική εξυπηρετεί αυτόν τον σκοπό. Νε μια προγραμματιζόμενη διεπαφή να υπάρχει στο standard υλικό ένα δίκτυο εξοπλισμένο από πολλαπλούς κατασκευαστές γίνεται πιθανότητα.

2.2.2 Scalability

Όσον αφορά την scalability(δυνατότητα κλιμάκωσης) μπορούμε να διαχωρίσουμε το θέμα σε scalability δικτύου και scalability κόμβων. Κυρίως επικεντρωνόμαστε στην scalability του controller. Σε αυτήν την περίπτωση έχουμε να αντιμετωπίσουμε τρείς προκλήσεις.

- Την καθυστέρηση λόγω της ανταλλαγής πληροφοριών δικτύου μεταξύ πολλών κόμβων και πολλών controller.
- Τον τρόπο επικοινωνίας του SDN controller με άλλους controllers μέσω των east και westbound APIs.
- Το μέγεθος και την λειτουργικότητα της βάσης δεδομένων του controller.

Στο δρόμο για να πετύχουμε την πλήρη scalability μια επαναστατική προσέγγιση της ικανότητας προγραμματισμού του δικτύου είναι απαραίτητης. Για παράδειγμα με την υβριδική αρχιτεκτονική ένας όγκος από queries είναι ικανό να λυθεί στην CPU του κόμβου, το οποίο ερώτημα σε αντίθετη περίπτωση θα έπρεπε να πάει στον controller για επεξεργασία. Αυτό μπορεί πιθανά να μειώσει το μέγεθος της βάσης δεδομένων στον controller και την ίδια στιγμή να μειώσει την επικοινωνία μεταξύ του controller και των άλλων κόμβων του.

2.2.3 Ασφάλεια

Υπάρχει μια περιορισμένη συζήτηση και έρευνα όσο αναφορά την ασφάλεια στα SDN. Μια ομάδα εργασίας που ασχολείται με την ασφάλεια υπάρχει στον Open Networking Foundation (ONF) . Τα κύρια προβλήματα ασφαλείας στο επίπεδο των εφαρμογών του controller είναι οι μηχανισμοί αυθεντικοποίησης και εξουσιοδότησης (authentication and authorization) ώστε να επιτρέψουν σε διάφορους οργανισμούς να έχουν πρόσβαση στους πόρους του controller, αλλά την ίδια στιγμή να παρέχουν την αντίστοιχη προστασία στους πόρους. Δεν απαιτούν όλες οι συσκευές τα ίδια προνόμια στο δίκτυο, για τον λόγο αυτό ένα μοντέλο ασφαλείας πρέπει να καθοριστεί για να απομονώνει εφαρμογές και να υποστηρίζει προστασία του δικτύου.

Μια πιθανή λύση είναι η εξουσιοδότηση να γίνεται βάσει ρόλου. Το FortNox προτείνεται για την επίλυση συγκρούσεων όταν ένας controller λαμβάνει συγκρουόμενους κανόνες ροών από δύο διαφορετικές εφαρμογές. Αυτό από μόνο του δεν αποτελεί πλήρη λύση του προβλήματος.

Οι controllers αποτελούν έναν ελκυστικό στόχο, καθώς μπορούν να αποτελέσουν single point of failure σημείο του δικτύου. Επιπλέον, με την απουσία στιβαρής και αφαλούς πλατφόρμας SDN, οι επιτιθέμενοι μπορούν να υποδυθούν τους controllers και να φέρουν εις πέρας κακόβουλες επιθέσεις.

Μια τεχνολογία όπως Transport Layer Security (TLS) μπορεί να εφαρμοστεί με αμοιβαία πιστοποίηση μεταξύ των controllers και των switches. Τα specs του OpenFlow καθορίζουν την χρήση του TLS. Παρόλα αυτά το χαρακτηριστικό της ασφάλειας είναι προαιρετικό και τα specifications για το TLS δεν έχουν καθοριστεί. Ολοκληρωμένες προδιαγραφές ασφαλείας μεταξύ του controller και του switch πρέπει να καθοριστούν ώστε να διασφαλίζεται η σύνδεση και να προστατεύονται τα δεδομένα που ανταλλάσσονται μεταξύ controller και switch.

Η χρήση του TLS είναι αρκετή αν ένας μόνο controller ελέγχει πολλά στοιχεία του δικτύου, αλλά όταν έχουμε πολλούς controllers να επικοινωνούν με ένα στοιχείο του δικτύου ή πολλαπλές λειτουργίες ελέγχου να επικοινωνούν με έναν κεντρικό ελεγκτή οι διαδικασίες πιστοποίησης και ελέγχου πρόσβασης γίνονται περίπλοκες. Η πιθανότητα κάποιος να αποκτήσει πρόσβαση χωρίς πιστοποίηση αυξάνεται και μπορεί να οδηγήσει σε χειραγώγηση του configuration των κόμβων ή της κίνησης που περνάει από τους κόμβους.

Ένα επιπλέον πρόβλημα είναι η χρήση open interfaces και γνωστών πρωτόκολλων. Τέτοια ζητήματα πρέπει να ληφθούν υπόψη όταν σχεδιάζεται η πλατφόρμα του SDN.

Κάποια από τα θετικά που παρέχει το SDN σε θέματα ασφαλείας είναι η ικανότητα να αλλάζει τις πολιτικές ασφαλείας που χρησιμοποιεί, η εισαγωγή καινούργιων υπηρεσιών ασφάλειας και η αξιολόγηση του δικτύου, ώστε να μπορεί να αναγνωρίσει έγκαιρα κάποια απειλή.

2.2.4 Πως μπορούν τα SDN να ενσωματωθούν στα σημερινά δίκτυα

Τα σημερινά δίκτυα εξυπηρετούν πολλά συστήματα και επιχειρήσεις και η πλήρης ανταλλαγή των υπαρχόντων δικτύων με SDN δεν είναι εφικτή. Αυτό μπορεί να συμβεί μόνο σε κλειστά περιβάλλοντα, όπως datacenters.

Η μετάβαση στο SDN απαιτεί ταυτόχρονη υποστήριξη του SDN καθώς και του υπάρχοντος υλικού. Το Path Computation Element (PCE) μπορεί να βοηθήσει στην σταδιακή μετάβαση στο SDN. Με το PCE το

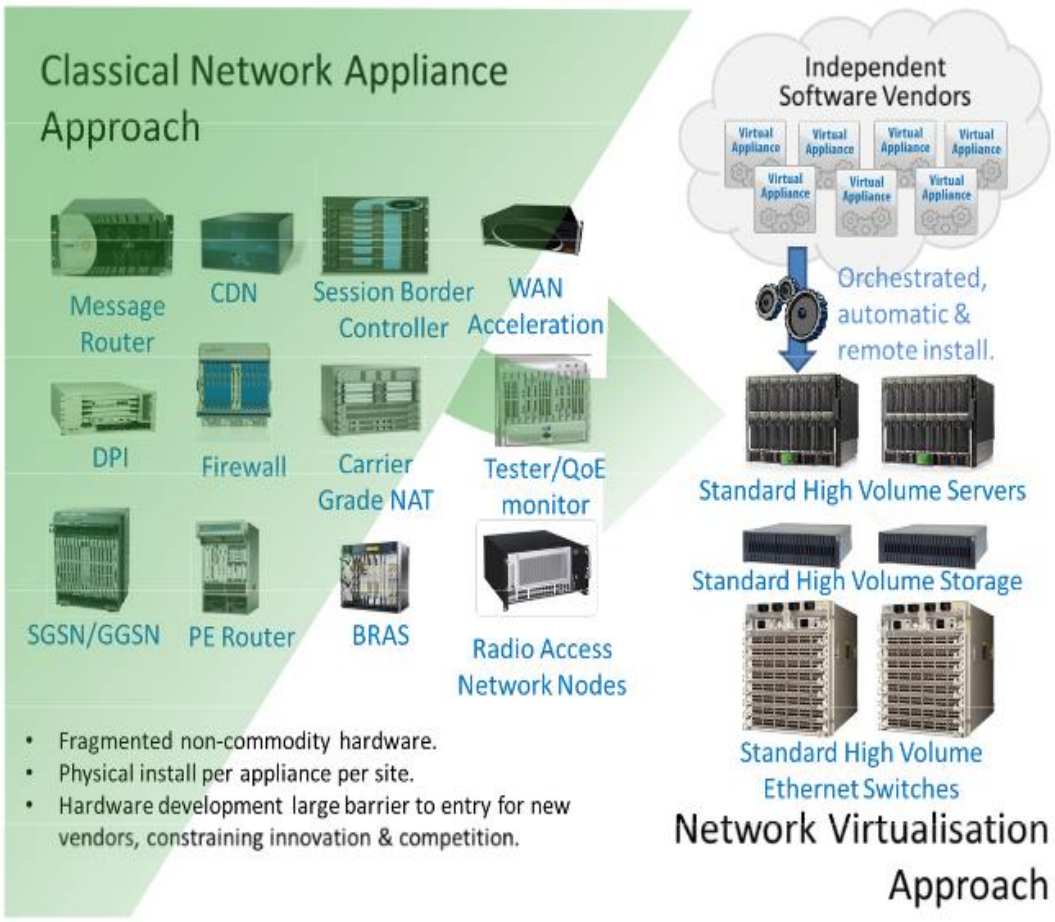
στοιχείο του δικτύου για τον υπολογισμό της διαδρομής αποκτά κεντρικοποιημένο ρόλο, ενώ τα παραδοσιακά στοιχεία του δικτύου που δεν χρησιμοποιούν PCE συνεχίζουν να υπολογίζουν τις διαδρομές με τον ίδιο τρόπο. Το πρωτόκολλο (PCE) επιτρέπει την επικοινωνία μεταξύ στοιχείων του δικτύου. Παρόλα αυτά το PCE Protocol - PCEP δεν παρέχει ολοκληρωτικό SDN, αλλά ο κεντρικοποιημένος SDN ελεγκτής παρέχει πλήρη υπολογισμό μονοπατιού για τις ροές διαμέσου πολλαπλών κόμβων του δικτύου.

Επιπλέον ανάπτυξη χρειάζεται για να επιτύχουμε ένα υβριδικό μοντέλο υποδομής SDN, στο οποίο παραδοσιακοί αλλά και υβριδικοί κόμβοι δικτύου μπορούν να λειτουργούν με αρμονία. Μια τέτοια λύση θα μείωνε τα κόστη, τα ρίσκα και την αποδιοργάνωση των επιχειρήσεων κατά την μετάβαση στην SDN υποδομή.

2.3 Network Functions Virtualization (NFV)

Ένα επιπλέον concept που έχει αναπτυχθεί είναι το Network Functions Virtualization (NFV), το οποίο προέρχεται από τους παρόχους υπηρεσιών οι οποίοι ήθελαν να επιταχύνουν την εφαρμογή των νέων υπηρεσιών δικτύου για να υποστηρίξουν τους στόχους και τα κέρδη τους. Οι περιορισμοί σε υλικό τους οδήγησε στην εφαρμογή των τεχνολογιών virtualization που χρησιμοποιούσαν στο IT να τις χρησιμοποιούν και στο δίκτυο τους. Στον οργανισμό ETSI Industry Specification Group for Network Functions Virtualization (ETSI ISG NFV), ένα group χρεώθηκε με την ανάπτυξη των απαιτήσεων και της αρχιτεκτονικής για το virtualization για διάφορες λειτουργίες μέσα στα τηλεπικοινωνιακά δίκτυα. Εάν είναι επιτυχημένο το NFV θα μειώσει τον όγκο του ιδιόκτητου υλικού που απαιτείται για να λειτουργήσουν οι υπηρεσίες δικτύου.

Τα μεγάλα πλεονεκτήματα του NFV είναι η μείωση τόσο των capital όσο και των operational expenses.

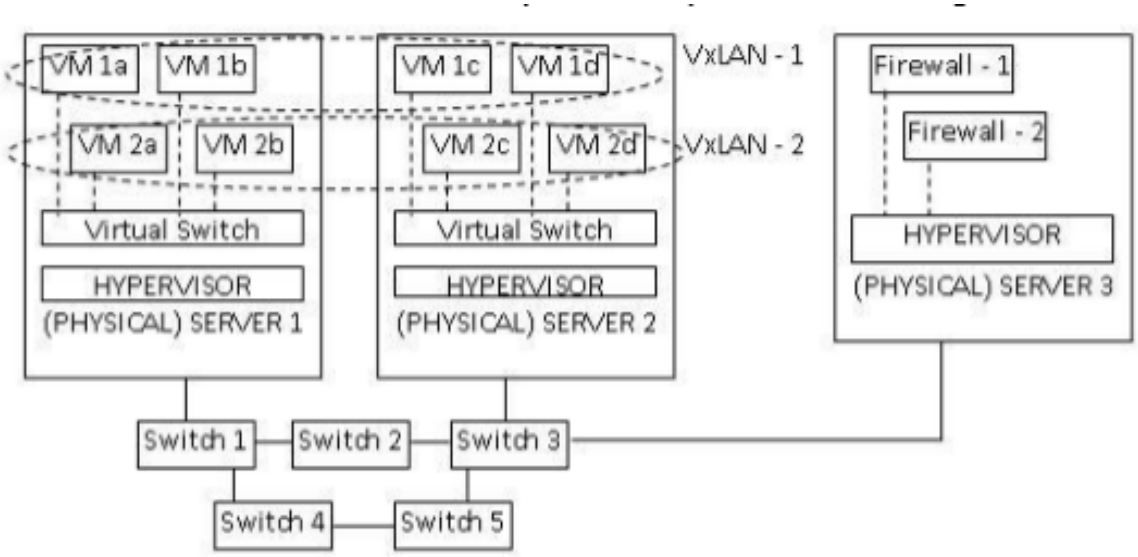


Εικόνα 2. 2: ***Άποψη το NFV***

2.3.1 Virtualization and SDN

Το NFV και το SDN είναι λύσεις που μπορούν να εφαρμοστούν και μόνες τους, αλλά ο συνδυασμός αυτών των δύο είναι πιο αποτελεσματικός. Η τεχνολογία NFV είναι ικανή να υποστηρίζει το SDN με το να παρέχει την υποδομή πάνω στη οποία το λογισμικό του SDN θα τρέχει. Επιπλέον, η τεχνολογία NFV στοιχίζεται με τους αντικειμενικούς σκοπούς του SDN για χρήση εμπορικών server και switches.Το SDN αποτελεί την μόνη πραγματική βιώσιμη απάντηση στις απαιτήσεις που θέτει το cloud computing , το virtualization των server καθώς και η αποθήκευση στα datacenters και στις επιχειρήσεων.

Η ευφυΐα του επιπέδου ελέγχου του δικτύου συγκεντρώνεται σε έναν ή λίγους servers και γίνεται ικανό να μεταφέρουμε τις ίδιες έννοιες του virtualization στο δίκτυο. Έτσι, κατά κάποιον τρόπο το network virtualization είναι πραγματοποιήσιμο μόνο στο SDN.

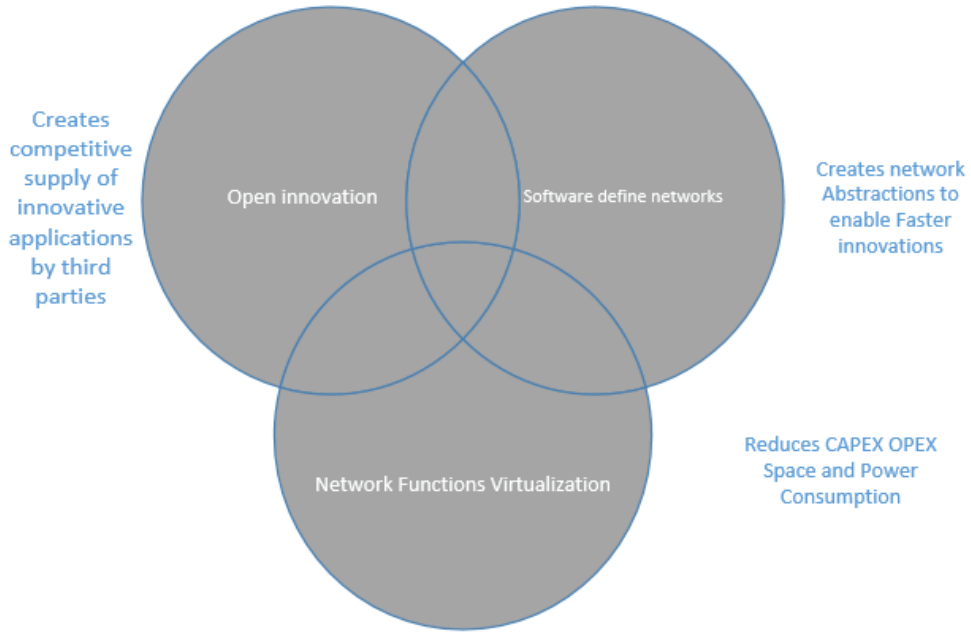


Εικόνα 2. 3: Δίκτυο που αποτελείται από πέντε switches και τρείς servers

Στην παραπάνω εικόνα μπορούμε να δούμε ένα δίκτυο που αποτελείται από πέντε switches και τρείς servers. Δύο από τους servers φιλοξενούν πολλαπλές virtual machines και ένα virtual switch. Ο τρίτος server φιλοξενεί προγράμματα firewall. Οι πόροι για την μια επιχείρηση 1 φιλοξενούνται δια μέσου των server, το ίδιο συμβαίνει και για την εταιρία 2. Τα virtual switches χρησιμοποιούνται για να εγκαθιδρύσουν την επιθυμητή σύνδεση μεταξύ των εικονικών μηχανών, δια μέσου των servers, μέσω των φυσικών switches. Τα φυσικά switches χρησιμοποιούνται μόνο για να παρέχουν την σύνδεση μεταξύ των server. Κάθε εταιρικό δίκτυο είναι ένα ξεχωριστό εικονικό δίκτυο πάνω από το φυσικό δίκτυο.

2.3.2 Διαφορές μεταξύ SDN και NFV

Ο κύριος στόχος του NFV είναι να μειώσει το κόστος εξοπλισμού και να παρέχει scalability, ελαστικότητα και ισχυρό οικοσύστημα. Το SDN όπως προτείνεται από τον Open Networking Foundation (ONF) σκοπεύει να επιτύχει τα ίδια πλεονεκτήματα. Ενώ το NFV έχει σκοπό να βελτιώσει την ανάπτυξη των λειτουργιών ενός network (όπως τα firewalls, DNS, loadbalancers, κτλ.), το OpenFlow SDN είναι περισσότερο συγκεντρωμένο στο routing και την βελτίωση του βαθύτερου δικτύου [44]. Όπως φαίνεται στη παρακάτω εικόνα, το NFV είναι συμπληρωματικό του SDN, αλλά όχι εξαρτώμενο από αυτό (ή το αντίθετο). Το NFV μπορεί να εφαρμοστεί χωρίς το SDN να απαιτείται, παρόλα αυτά οι δύο λύσεις μπορών να συνδυαστούν για να έχουμε το μέγιστο αποτέλεσμα.



Εικόνα 2. 4: Network Functions Virtualization Σχέση με το SDN, πηγή[4]

2.4 OpenFlow Πρωτόκολλο

Το πρωτόκολλο OpenFlow υποστηρίζει τρείς τύπους μηνυμάτων, controller-to-switch, ασυγχρονα, και συμμετρικά, κάθε ένα με πολλαπλούς υπο-τύπους. Τα μηνύματα controller-to-switch ξεκινούν από τον controller και χρησιμοποιούνται για να διαχειρίζονται ή να ελέγχουν την κατάσταση του switch.
Τα ασύγχρονα μηνύματα ξεκινούν από το switch και χρησιμοποιούνται για να ενημερώσουν τον controller για γεγονότα του δικτύου και για αλλαγές στην κατάσταση του switch.
Αντίστοιχα, τα συμμετρικά (symmetric) μηνύματα ξεκινούν είτε από το switch ή τον controller και αποστέλλονται χωρίς κάποια αίτηση.
Οι τύποι μηνυμάτων που χρησιμοποιούνται από το Openflow περιγράφονται παρακάτω

- Controller-προς-Switch

Τα Controller/switch μηνύματα ξεκινούν από τον ελεγκτή και μπορεί να χρειάζονται ή όχι απάντηση από το switch.

- Χαρακτηριστικά:

Ένας ελεγκτής μπορεί να απαιτήσει τις ιδιότητες ενός switch στέλνοντας ένα features request, το switch πρέπει να απάντησε με features reply που καθορίζει τις ιδιότητες του switch. Αυτό συνήθως πραγματοποιείται κατά την εγκαθίδρυση του OpenFlow καναλιού.

- Ρυθμίσεις:

Ο ελεγκτής είναι ικανός να τοποθετεί και να ρωτάει(query) τις παραμέτρους του configuration στο switch.Το switch απαντάει μόνο σε ερωτήσεις(query) από τον ελεγκτή.

- Modify-State:

Μηνύματα Modify-State στέλνονται από τον ελεγκτή για την διαχείριση της κατάστασης στα switches.

Ο πρωταρχικός σκοπός είναι να προσθέσει, διαγράψει ή να ενημερώσει groups στους OpenFlow πίνακες και να τοποθετήσει ιδιότητες στις πόρτες του switch.

- Read-State:

Τα Read-State μηνύματα χρησιμοποιούνται από τον controller για να συλλέξει στατιστικά από το switch.

- Packet-out:

Αυτά χρησιμοποιούνται από τον controller για να στείλουν πακέτα προς μια καθορισμένη πόρτα στο switch, και να προωθήσουν πακέτα που λαμβάνονται από τα Packet-in μηνύματα. Τα Packet-out πρέπει να περιέχουν ένα πλήρες πακέτο ή μια ID που αναφέρεται σε πακέτο το οποίο είναι αποθηκευμένο στο switch. Το μήνυμα πρέπει επίσης να περιέχει μια λίστα από ενέργειες οι ποίες πρέπει να εφαρμοστούν όπως έχουν καθοριστεί, μια κενή λίστα ενεργειών ρίχνει το πακέτο.

- Barrier:

Τα Barrier request/reply μηνύματα χρησιμοποιούνται από τον controller για να διασφαλίσουν ότι εξαρτήσεις των μηνυμάτων έχουν εξασφαλιστεί ή για την λήψη προειδοποιήσεων για ολοκληρωμένες ενέργειες.

- Role-Request:

Τα Role-Request μηνύματα χρησιμοποιούνται από τον controller για να δώσουν το ρόλο του καναλιού OpenFlow ή για να ρωτήσουν για τον ρόλο. Αυτό είναι περισσότερο χρήσιμο όταν το switch συνδέεται σε πολλαπλούς controllers.

- Asynchronous-Configuration:

Οι ρυθμίσεις για τα ασύγχρονα μηνύματα χρησιμοποιούνται από τον controller για να τοποθετήσουν επιπλέον φίλτρο στα ασύγχρονα μηνύματα που θέλει να λάβει ο ελεγκτής στο OpenFlow κανάλι, ή να ρωτήσει για αυτό το φίλτρο. Αυτό είναι χρήσιμο όταν το switch συνδέεται σε πολλαπλούς controllers, και συχνά εκτελείται σε ένα εγκαθιδρυμένο OpenFlow κανάλι.

- Asynchronous

Τα Asynchronous μηνύματα στέλνονται χωρίς ο controller να τα ζητήσει από το switch.
Τα Switches στέλνουν ασύγχρονα μηνύματα στον ελεγκτή για να υποδηλώσει την άφιξη πακέτου, ή την αλλαγή κατάστασης του switch. Οι τέσσερις κύριοι τύποι ασύγχρονων μηνυμάτων περιγράφονται παρακάτω.

- Packet-in:

Μεταφέρει όλο τον έλεγχο ενός πακέτου στον controller.Για όλα τα πακέτα που προωθούνται στην δεσμευμένη πόρτα του ελεγκτή χρησιμοποιώντας μια flow entry ή μια flow entry που οδήγησε σε αστοχία πίνακα, ένα packet-in γεγονός στέλνεται στους controllers. Περαιτέρω επεξεργασία, όπως TTL έλεγχος, 0μπορεί επίσης να παράγει packet-in γεγονότα για να στέλνονται πακέτα στον controller.
Για γεγονότα packet-in που παράγονται από εξωτερικές ενέργειες σε flow entries, το packet-in μπορεί να καθορίζεται ατομικά στην εξωτερική ενέργεια, άλλα packet-in μπορούν να ρυθμιστούν στις switch configuration. Εάν τα packet-in ρυθμίζονται σε buff er πακέτα και το switch έχει επαρκή μνήμη για να τα κάνει buffer, τα packet-in γεγονότα περιέχουν μόνο κάποιο τμήμα της επικεφαλίδας του πακέτου και μια buff er ID , τα οποία χρησιμοποιούνται από τον controller όταν το switch είναι έτοιμο να προωθήσει τα πακέτα. Τα switches που δεν υποστηρίζουν εσωτερικό buff ering, ρυθμίζονται έτσι ώστε να μην κρατούν στον buff er τα πακέτα για τα packet-in γεγονότα, αλλά πρέπει να στείλουν τα πακέτα στους controllers σαν μέρος της ενέργειας. Τα πακέτα που βρίσκονται στο buffer θα επεξεργαστούν μέσω των Packet-out μηνυμάτων από τον controller, ή θα λήξουν μετά από κάποιο χρόνο.
Εάν κάποιο πακέτο βρίσκεται στον buff er, ο αριθμός από bytes του αυθεντικού πακέτου που συμπεριλαμβάνεται στο packet-in μπορεί να ρυθμιστεί. By default, είναι 128 bytes.

- Flow-Removed:

Όταν μια flow entry προστίθεται σε ένα switch από ένα flow modify μήνυμα, μια τιμή idle timeout δείχνει πότε μια flow entry πρέπει να αφαιρείται επειδή δεν υπάρχει κινητικότητα για αυτήν την flow entry, επίσης υπάρχει και τιμή για hard timeout η οποία δείχνει ότι ένα flow entry πρέπει να αφαιρεθεί ανεξαρτήτως κινητικότητας. Το μήνυμα flow modify επίσης καθορίζει πότε το switch πρέπει να στείλει flow removed μήνυμα στον controller όταν το flow λήγει. Flow delete αιτήσεις πρέπει να δημιουργούνται όταν τα flow removed μηνύματα για οποιαδήποτε flows έχουν σεταρισμένο το tag OFPFF_SEND_FLOW_REM.

- Port-status:

Ενημερώνει τον controller για αλλαγή σε μια πόρτα. Το switch πρέπει να στείλει port-status μηνύματα στους controllers σαν μέρος των ρυθμίσεων ή επειδή έχει αλλάξει η κατάσταση μιας πόρτας. Αυτά τα γεγονότα περιλαμβάνουν αλλαγή στις ρυθμίσεις της πόρτας, για παράδειγμα εάν η πόρτα έκλεισε από τον χρήστη, και γεγονότα αλλαγής κατάστασης της πόρτας, για παράδειγμα εάν η ζεύξη έπεσε.

- Error:

Το switch είναι ικανό να ενημερώνει τον controller για προβλήματα χρησιμοποιώντας μηνύματα λάθους.

- Symmetric

Τα symmetric στέλνονται χωρίς προειδοποίηση προς τις δύο κατευθύνσεις.

- Hello:

Hello μηνύματα ανταλλάσσονται μεταξύ του switch και του controller κατά την αρχή της σύνδεσης.

- Echo:

Τα Echo request/reply μηνύματα μπορούν να στέλνονται είτε από το switch ή από τον controller, και πρέπει να επιστρέφουν echo reply. Μπορούν να χρησιμοποιηθούν για να μετρήσουμε την καθυστέρηση ή το bandwidth μιας σύνδεσης μεταξύ controller και switch, όπως επίσης αν είναι ενεργοί οι κόμβοι.

- Experimenter:

Τα Experimenter παρέχουν ένα τρόπο στα OpenFlow switches να προσφέρουν επιπλέον λειτουργικότητα μέσα στον χώρο του OpenFlow τύπου μηνύματος.

- Message Handling

Το OpenFlow παρέχει αξιόπιστη παράδοση και επεξεργασία μηνυμάτων, αλλά δεν παρέχει acknowledgements ή εγγυημένη επεξεργασία μηνυμάτων με την σωστή σειρά. Η συμπεριφορά της OpenFlow διαχείρισης μηνυμάτων που περιγράφεται σε αυτό το τμήμα παρέχεται στην κύρια και βοηθητική σύνδεση χρησιμοποιώντας αξιόπιστη μεταφορά.

- Message Delivery:

Υπάρχει εγγύηση για την παράδοση των μηνυμάτων, εκτός εάν ολόκληρο το OpenFlow κανάλι αποτύχει, σε αυτήν την περίπτωση ο controller δεν πρέπει να υποθέτει τίποτα σχετικά με την κατάσταση του switch (π.χ., το switch μπορεί να έχει μπει σε "fail standalone mode").

- Message Processing:

Τα Switches πρέπει να επεξεργάζονται κάθε μήνυμα που λαμβάνουν από τον controller πλήρως, και πιθανά να δημιουργήσουν ια απάντηση. Εάν ένα switch δεν μπορεί πλήρως να επεξεργαστεί ένα μήνυμα που λαμβάνει από τον controller, πρέπει να στείλει πίσω ένα μήνυμα λάθους. Για τα packet-out μηνύματα, η πλήρης επεξεργασία του μηνύματος δεν εγγυάται ότι το πακέτο υπάρχει στο switch. Το πακέτο μπορεί να έχει αφαιρεθεί μετά την OpenFlow επεξεργασία εξαιτίας της συμφόρησης στο switch,την πολιτική ποιότητας(QoS policy),ή εάν έχει σταλεί σε μια μπλοκαρισμένη ή μη έγκυρη πόρτα.
Επιπρόσθετα, τα switches πρέπει να στέλνουν στον controller όλα τα ασύγχρονα μηνύματα που δημιουργούνται από τις OpenFlow αλλαγές κατάστασης, όπως τα flow-removed, port-status ή packet-in μηνύματα, έτσι ώστε η οπτική του controller για το switch να είναι ακριβής με την κατάστασή του. Αυτά τα μηνύματα μπορεί να φιλτράρονται βασιζόμενα στις Asynchronous ρυθμίσεις. Επιπλέον, καταστάσεις

που μπορούν να πυροδοτήσουν μια OpenFlow αλλαγή κατάστασης μπορεί να φιλτραριστούν πριν ακόμα μπορέσουν να κάνουν αυτήν την αλλαγή. Για παράδειγμα, τα πακέτα που λαμβάνονται στις πόρτες δεδομένων και πρέπει να προωθηθούν στον controller μπορεί να αφαιρεθούν εξαιτίας της συμφόρησης ή της πολιτική ποιότητας στο switch και να μην παράγουν packet-in μηνύματα. Αυτές οι απορρίψεις μπορούν να συμβούν για πακέτα με μια συγκεκριμένη ενέργεια εξόδου προς τον controller. Αυτές οι απορρίψεις μπορεί να συμβούν όταν κάποιο πακέτο δεν ταιριάξει με κάποιο από τις entries στο πίνακα και η προκαθορισμένη ενεργεί του πίνακα στέλνεται στον controller. Η πολιτική των πακέτων που προορίζεται για τον controller χρησιμοποιώντας QoS ενέργειες είναι περιορισμένη, για να αποφύγουμε απόρριψη υπηρεσιών από τον controller connection. Οι controllers μπορούν ελεύθερα να αγνοήσουν μηνύματα που λαμβάνουν, αλλά πρέπει να απαντήσουν σε echo μηνύματα για να αποτρέψουν το switch από το να τερματίσει την σύνδεση.

- Message Ordering:

Η ακολουθία μπορεί να επιτευχθεί μέσω της χρήσης barrier μηνυμάτων. Στην απουσία των barrier μηνυμάτων, τα switches μπορεί τυχαία να αλλάξουν την διάταξη των μηνυμάτων για να βελτιώσουν την απόδοση ως εκ τούτου οι controllers δεν πρέπει να βασίζονται σε συγκεκριμένη σειρά επεξεργασίας. Συγκεκριμένα, οι flow entries μπορούν να τοποθετηθούν στον πίνακα με μια σειρά διαφορετική από την ροή των μηνυμάτων που λαμβάνονται από το switch. Τα μηνύματα δεν πρέπει να επαναδιατάσσονται διαμέσου ενός barrier message και το barrier message πρέπει να επεξεργαστεί μόνο όταν τα προηγούμενα μηνύματα έχουν επεξεργαστεί.

- Για την ακρίβεια:

1.Μηνύματα πριν από έναν barrier πρέπει να επεξεργαστούν πλήρως, συμπεριλαμβανομένου της αποστολής και της αντίστοιχης απάντησης ή λαθών
2. Το barrier πρέπει να επεξεργάζεται και μια barrier απάντηση να στέλνεται
3. Τα μηνύματα μετά τον barrier μπορούν να ξεκινήσουν να επεξεργάζονται
Εάν δύο μηνύματα από τον controller βασίζονται το ένα στο άλλο (π.χ. μια flow mod προσθήκη με το αντίστοιχο packet-out προς OFPP_TABLE), πρέπει να διαχωρίζονται από ένα barrier μήνυμα.

- Συνδέσεις των καναλιών OpenFlow

Το OpenFlow χρησιμοποιείται για ανταλλαγή OpenFlow μηνυμάτων μεταξύ ενός OpenFlow switch και ενός OpenFlow controller. Ένας τυπικός OpenFlow controller διαχειρίζεται πολλαπλά OpenFlow κανάλια, κάθε ένα σε διαφορετικό OpenFlow switch. Ένα OpenFlow switch μπορεί να έχει ένα OpenFlow κανάλι σε έναν controller, ή πολλαπλά κανάλια για αξιοπιστία, το καθένα σε διαφορετικό controller.
Ένας OpenFlow controller τυπικά διαχειρίζεται έναν OpenFlow switch απομακρυσμένα πάνω από ένα ή πολλά δίκτυα.
Το OpenFlow κανάλι συχνά αρχικοποιείται σαν μια απλή σύνδεση δικτύου, χρησιμοποιώντας TLS ή απλό TCP. Το OpenFlow μπορεί να αποτελείται από πολλαπλές συνδέσεις δικτύου ώστε να εκμεταλλευτεί την παραλληλία. Το OpenFlow switch πάντα ξεκινάει τις συνδέεις προς τον OpenFlow controller.

- Σετάρισμα συνδέσεων

Το switch πρέπει να είναι ικανό να εγκαθιδρύσει την επικοινωνία με ένα controller σε μια ρυθμιζόμενη από τον χρήστη IP διεύθυνση, χρησιμοποιώντας μια πόρτα καθορισμένη από τον χρήστη. Εάν το switch ξέρει την IP διεύθυνση του controller, το switch ξεκινάει μια σύνδεση μέσω TLS ή TCP με τον controller. Η κίνηση είναι από και προς το OpenFlow δεν τρέχει διαμέσου του OpenFlow αγωγού. Για τον λόγο αυτό, το switch πρέπει να αναγνωρίσει τη εισερχόμενη κίνηση σαν τοπική πριν την τσεκάρει με τα flow tables. Όταν μια OpenFlow σύνδεση εγκαθίσταται για πρώτη φορά, κάθε πλευρά της σύνδεσης πρέπει να στείλει ένα μήνυμα OFPT_HELLO με το πεδίο της έκδοσης σεταρισμένο στην υψηλότερη έκδοση του πρωτοκόλλου OpenFlow που υποστηρίζεται από τον αποστολέα. Αυτό το Hello μήνυμα μπορεί επιλεκτικά να συμπεριλαμβάνει κάποια OpenFlow στοιχεία για να βοηθήσουν το σετάρισμα της σύνδεσης. Κατά την λήψη αυτού του μηνύματος, ο παραλήπτης πρέπει να υπολογίσει τη έκδοση του OpenFlow πρωτοκόλλου που θα χρησιμοποιηθεί. Εάν το Hello μήνυμα που στέλνεται αλλά και το Hello μήνυμα που λαμβάνεται περιλαμβάνουν ένα OFPHET_VERSIONBITMAP hello στοιχείο, και αν αυτά τα bitmaps έχουν μερικά

κοινά bits σεταρισμένα, η έκδοση πρέπει να είναι η υψηλότερη και στα δύο bitmaps. Εάν η έκδοση υποστηρίζεται από τον παραλήπτη τότε η σύνδεση συνεχίζεται. Αλλιώς ο παραλήπτης πρέπει να απαντήσει με ένα μήνυμα OFPT_ERROR με πεδίο type OFPET_HELLO_FAILED, πεδίο code OFPHFC_INCOMPATIBLE, και επιλεκτικά ένα ASCII string εξηγώντας την κατάσταση, και μετά να τερματίσει την σύνδεση.

- Διακοπή Συνδέσεων

Στην περίπτωση που ένα switch χάσει επαφή με τον controller, σαν αποτέλεσμα ενός echo request timeout,TLS session timeout, ή άλλη αιτίας αποσύνδεσης, πρέπει να προσπαθήσει να συνδεθεί με έναν ή περισσότερους εναλλακτικούς controllers.
Η σειρά με την οποία το switch επικοινωνεί με τους εφεδρικούς controllers δεν καθορίζεται από το πρωτόκολλο.
Το switch πρέπει αμέσως να εισάγει είτε την “fail secure mode” είτε την “fail standalone mode” εάν χάσει την σύνδεση με τονcontroller, σύμφωνα με την εφαρμογή του switch αλλά και με τις ρυθμίσεις. Σε “fail secure mode”, η μόνη αλλαγή στην συμπεριφορά του switch είναι ότι τα πακέτα και τα μηνύματα που προορίζονται για τον controller που έχει πέσει, απορρίπτονται. Τα flows πρέπει να συνεχίσουν να λήγουν σύμφωνα με τα timeouts σε “fail secure mode”. Σε "fail standalone mode", το switch επεξεργάζεται όλα τα πακέτα χρησιμοποιώντας την OFPP_NORMAL πόρτα, με άλλα λόγια το switch συμπεριφέρεται σαν Ethernet switch ή router.
Κατά την επανασύνδεση με έναν controller η υπάρχουσα flow entry παραμένει. Ο controller τότε έχει την επιλογή να διαγράψει όλα τα flow entries, εάν το επιθυμεί.
Την πρώτη φορά που το switch ξεκινά, θα εκτελέσει είτε την "fail secure mode" ή την "fail standalone mode". Οι ρυθμίσεις που θα χρησιμοποιηθούν κατά την εκκίνηση είναι εκτός του
Περιθωρίου του OpenFlow πρωτοκόλλου.

- Πολλαπλοί Ελεγκτές

Το switch μπορεί να εγκαθιδρύσει επικοινωνία με έναν μόνο controller, ή μπορεί να εγκαθιδρύσει επικοινωνία με πολλούς controllers. Έχοντας πολλαπλούς controllers βελτιώνεται η αξιοπιστία, καθώς το switch μπορεί να συνεχίζει να λειτουργεί σε OpenFlow mode εάν η σύνδεση σε έναν ελεγκτή αποτύχει. Η παράδοση μεταξύ των controllers διαχειρίζεται πλήρως από τους controllers το οποίο επιτρέπει γρήγορη ανάκαμψη από αποτυχίες καθώς και controller load balancing. Η λειτουργιά των πολλαπλών controller διευθυνσιοδοτεί μόνο θέματα που έχουν να κάνουν με το fail-over και το load balancing του controller, και όχι θέματα virtualization τα οποία μπορούν να γίνουν εκτός του OpenFlow πρωτοκόλλου.

Όταν αρχικοποιείται η λειτουργία του OpenFlow, το switch πρέπει να συνδεθεί σε όλους τους controllers που είναι ρυθμισμένο να συνδεθεί, και να προσπαθήσει να διατηρήσει την συνδεσιμότητα με όλους ταυτόχρονα. Πολλοί controllers μπορούν να στέλνουν controller προς-switch εντολές, η απάντηση ή το μήνυμα λάθους σε αυτές τις εντολές πρέπει μόνο να στέλνεται μόνο στην σύνδεση του controller που σχετίζεται με αυτές τις εντολές. Τα ασύγχρονα μηνύματα μπορεί επίσης να χρειαστούν να σταλούν σε πολλαπλούς controllers, το μήνυμα γίνεται διπλό για κάθε OpenFlow κανάλι και κάθε μήνυμα στέλνεται όταν η αντίστοιχη σύνδεση στον controller το επιτρέψει.
Ο προκαθορισμένος ρόλος του controller είναι ο OFPCR_ROLE_EQUAL. Σε αυτόν τον ρόλο, ο controller πρέπει ν α έχει πλήρη πρόσβαση στο switch και είναι ίσο με άλλους controllers που βρίσκονται στον ίδιο ρόλο. Ο controller λαμβάνει όλα τα σύγχρονα μηνύματα του switch (όπως τα packet-in, flow-removed). Ο controller μπορεί να στείλει controller-προς-switch εντολές για να αλλάξει την κατάσταση του switch. Το switch δεν κάνει καμιά αυθαιρεσία ή μοίρασμα πόρων μεταξύ των controllers.
Ένας controller μπορεί να απαιτήσει ο ρόλος του να αλλάξει σε OFPCR_ROLE_SLAVE. Σε αυτόν το ρόλο ο controller πρόσβαση μόνο για διάβασμα στο switch. Σε προκαθορισμένη κατάσταση ο controller δεν λαμβάνει ασύγχρονα μηνύματα στο switch, εκτός από μηνύματα για την κατάσταση της πόρτας. Ο controller δεν εκτελεί εντολές από τον controller-στο-switch οι οποίες τροποποιούν την κατάσταση του switch όπως οι, OFPT_PACKET_OUT, OFPT_FLOW_MOD, OFPT_GROUP_MOD, OFPT_PORT_MOD και OFPT_TABLE_MOD. Εάν ο controller στείλει μια από αυτές τις εντολές, το switch πρέπει να απαντήσει με ένα OFPT_ERROR μήνυμα με το πεδίο τύπου να έχει τιμή OFPET_BAD_REQUEST, το πεδίο code να έχει τιμή OFPBRC_IS_SLAVE. Άλλα controller-προς-switch

μηνύματα, όπως το OFPT_MULTIPART_REQUEST και OFPT_ROLE_REQUEST, πρέπει να επεξεργάζονται κανονικά.
Ένας controller μπορεί να απαιτήσει ο ρόλος του να αλλαχτεί σε OFPCR_ROLE_MASTER. Αυτός ο ρόλος είναι παρόμοιος με τον OFPCR_ROLE_EQUAL και έχει πλήρη πρόσβαση στο switch, η διαφορά είναι ότι το switch σιγουρεύει ότι είναι ο μόνος controller σε αυτόν τον ρόλο. Όταν ένας controller αλλάξει τον ρόλο του σε OFPCR_ROLE_MASTER, το switch αλλάζει όλους τους άλλους controllers με ρόλο OFPCR_ROLE_MASTER ώστε να έχουν το ρόλο OFPCR_ROLE_SLAVE. Όταν το switch εκτελέσει μια τέτοια αλλαγή ρόλων, κανένα μήνυμα δεν παράγεται στον controller του οποίου ο ρόλος έχει αλλάξει (στις περισσότερες περιπτώσεις που ο controller δεν είναι πια προσβάσιμος).
Ένα switch μπορεί να είναι ταυτόχρονα συνδεδεμένο σε πολλαπλούς controllers σε Equal κατάσταση, πολλαπλοί controllers σε Slave κατάσταση, και τουλάχιστον ένας controller σε Master σκατάσταση. Κάθε controller μπορεί να στέλνει ένα OFPT_ROLE_REQUEST μήνυμα για να επικοινωνεί τον ρόλο του στο switch, και το switch πρέπει να θυμάται τον ρόλο του κάθε controller. Ένας controller μπορεί να αλλάζει ρόλο κάθε στιγμή.
Ένας controller μπορεί επίσης να ελέγχει ποιοί τύποι ασύγχρονων μηνυμάτων του switch μπορούν να στέλνονται μέσω του OpenFlow καναλιού, και να αλλάζει τις προκαθορισμένες τιμές που αναφέρθηκαν παραπάνω. Αυτό γίνεται μέσω Asynchronous Configuration μηνυμάτων, τα οποία τοποθετούν σε λίστα όλους του λόγους, για κάθε τύπο μηνύματος, το οποίο πρέπει να φιλτραριστεί ή ενεργοποιηθεί για το συγκεκριμένο OpenFlow κανάλι. Χρησιμοποιώντας αυτό το χαρακτηριστικό, διαφορετικοί controllers μπορούν να λαμβάνουν διαφορετικές ενημερώσεις, ένας controller σε master mode μπορεί επιλεκτικά να απενεργοποιήσει τις ενημερώσεις για τις οποίες δεν ενδιαφέρεται και ένας controller σε slave mode μπορεί να ενεργοποιήσει τις ενημερώσεις τις οποίες επιθυμεί.
Για την ανεύρεση των μηνυμάτων που είναι εκτός σειράς κατά την μετάβαση από κατάσταση master/slave, το μήνυμα OFPT_ROLE_REQUEST περιλαμβάνει ένα 64-bit sequence number πεδίο, το generation_id, το οποίο αναγνωρίζει την mastership ιδιότητα. Σαν μέρος του μηχανισμού εκλογής του master, οι controllers συντονίζουν την ανάθεση του generation_id. Το generation_id είναι ένας μετρητής που αυξάνεται μονοτονικά: ένα καινούργιο generation_id τοποθετείται κάθε φορά που το mastership view αλλάζει, π.χ. όταν ένας καινούργιος master is ορίζεται.
Κατά την λήψη ενός OFPT_ROLE_REQUEST με ρόλο OFPCR_ROLE_MASTER ή OFPCR_ROLE_SLAVE το switch πρέπει να συγκρίνει το generation_id στο μήνυμα σύμφωνα με την μεγαλύτερη generation id που είχε έως τώρα. Ένα μήνυμα με generation_id μικρότερο από κάποιο προηγούμενο generation id πρέπει να θεωρείται παλιό και να διαγράφεται. Το switch πρέπει να απαντάει στο παλιό μήνυμα με ένα μήνυμα λάθους με τύπο OFPET_ROLE_REQUEST_FAILED και code OFPRRFC_STALE.

2.5 Openflow Analysis

Η καρδιά ενός Openflow switch είναι το openflow πρωτόκολλο.

Η struct που καθορίζει την επικεφαλίδα του μηνύματος openflow είναι το παρακάτω

```
/* Header on all OpenFlow packets. */
struct ofp_header {
uint8_t version; /* OFP_VERSION. */
uint8_t type; /* One of the OFPT_ constants. */
uint16_t length; /* Length including this ofp_header. */
uint32_t xid; /* Transaction id associated with this packet.
Replies use the same id as was in the request
to facilitate pairing. */
};
OFP_ASSERT(sizeof(struct ofp_header) == 8);
```

Το πεδίο version είναι ένα byte και καθορίζει την έκδοση του πρωτοκόλλου που χρησιμοποιείται. Το length με μήκος δύο byte είναι το συνολικό μήκος του μηνύματος. Το xid είναι το transaction id που σχετίζεται με το συγκεκριμένο πακέτο και τα replies πρέπει να έχουν το ίδιο transaction id με τα requests.

Το πεδίο type μπορεί να έχει τις παρακάτω τιμές:

```
enum ofp_type {
/* Immutable messages. */
OFPT_HELLO = 0, /* Symmetric message */
OFPT_ERROR = 1, /* Symmetric message */
OFPT_ECHO_REQUEST = 2, /* Symmetric message */
OFPT_ECHO_REPLY = 3, /* Symmetric message */
OFPT_EXPERIMENTER = 4, /* Symmetric message */
/* Switch configuration messages. */
OFPT_FEATURES_REQUEST = 5, /* Controller/switch message */
OFPT_FEATURES_REPLY = 6, /* Controller/switch message */
OFPT_GET_CONFIG_REQUEST = 7, /* Controller/switch message */
OFPT_GET_CONFIG_REPLY = 8, /* Controller/switch message */
OFPT_SET_CONFIG = 9, /* Controller/switch message */
/* Asynchronous messages. */
OFPT_PACKET_IN = 10, /* Async message */
OFPT_FLOW_REMOVED = 11, /* Async message */
OFPT_PORT_STATUS = 12, /* Async message */
/* Controller command messages. */
OFPT_PACKET_OUT = 13, /* Controller/switch message */
OFPT_FLOW_MOD = 14, /* Controller/switch message */
OFPT_GROUP_MOD = 15, /* Controller/switch message */
OFPT_PORT_MOD = 16, /* Controller/switch message */
OFPT_TABLE_MOD = 17, /* Controller/switch message */
/* Multipart messages. */
OFPT_MULTIPART_REQUEST = 18, /* Controller/switch message */
OFPT_MULTIPART_REPLY = 19, /* Controller/switch message */
/* Barrier messages. */
OFPT_BARRIER_REQUEST = 20, /* Controller/switch message */
OFPT_BARRIER_REPLY = 21, /* Controller/switch message */
/* Controller role change request messages. */
OFPT_ROLE_REQUEST = 24, /* Controller/switch message */
OFPT_ROLE_REPLY = 25, /* Controller/switch message */
/* Asynchronous message configuration. */
OFPT_GET_ASYNC_REQUEST = 26, /* Controller/switch message */
OFPT_GET_ASYNC_REPLY = 27, /* Controller/switch message */
OFPT_SET_ASYNC = 28, /* Controller/switch message */
/* Meters and rate limiters configuration messages. */
OFPT_METER_MOD = 29, /* Controller/switch message */
/* Controller role change event messages. */
OFPT_ROLE_STATUS = 30, /* Async message */
/* Asynchronous messages. */
OFPT_TABLE_STATUS = 31, /* Async message */
/* Request forwarding by the switch. */
OFPT_REQUESTFORWARD = 32, /* Async message */
/* Bundle operations (multiple messages as a single operation). */
OFPT_BUNDLE_CONTROL = 33,
OFPT_BUNDLE_ADD_MESSAGE = 34,
};
```

2.5.1 Δομή των Ports

Το openflow λαμβάνει και στέλνει μηνύματα σε πόρτες. Το switch μπορεί να καθορίζει φυσικές και λογικές πόρτες, και τα specs του openflow καθορίζουν μερικές δεσμευμένες πόρτες.

Οι φυσικές πόρτες, οι λογικές πόρτες και οι δεσμευμένες πόρτες καθορίζονται από την δομή ofp_port, που περιγράφεται παρακάτω.

```
/* Description of a port */

 struct ofp_port {

 uint32_t port_no;

uint8_t pad[4];

 uint8_t hw_addr[OFP_ETH_ALEN];

uint8_t pad2[2]; /* Align to 64 bits. */

 char name[OFP_MAX_PORT_NAME_LEN]; /* Null-terminated */

uint32_t config; /* Bitmap of OFPPC_* flags. */

uint32_t state; /* Bitmap of OFPPS_* flags. */

/* Bitmaps of OFPPF_* that describe features. All bits zeroed if * unsupported or unavailable. */

uint32_t curr; /* Current features. */

 uint32_t advertised; /* Features being advertised by the port. */

 uint32_t supported; /* Features supported by the port. */

uint32_t peer; /* Features advertised by peer. */

uint32_t curr_speed; /* Current port bitrate in kbps. */

 uint32_t max_speed; /* Max port bitrate in kbps */

}; OFP_ASSERT(sizeof(struct ofp_port) == 64);
```

Το πεδίο port_no αναγνωρίζει μοναδικά την πόρτα μέσα στο switch. Το πεδίο hw_addr είναι η MAC διεύθυνση για την πόρτα. Το OFP_MAX_ETH_ALEN είναι 6. Το πεδίο name είναι ένα string που περιέχει το όνομα για το interface. Η τιμή του OFP_MAX_PORT_NAME_LEN είναι 16.

Το πεδίο config περιγράφει τις ρυθμίσεις που μπορούμε να δώσουμε σαν διαχειριστές, και έχει την παρακάτω δομή

```
/* Flags to indicate behavior of the physical port. These flags are * used in ofp_port to describe the current configuration. They are * used in the ofp_port_mod message to configure the port's behavior. */

 enum ofp_port_config {
```

OFPPC_PORT_DOWN = 1 << 0, /* Port is administratively down. */

OFPPC_NO_RECV = 1 << 2, /* Drop all packets received by port. */

OFPPC_NO_FWD = 1 << 5, /* Drop packets forwarded to port. */

OFPPC_NO_PACKET_IN = 1 << 6 /* Do not send packet-in msgs for port. */

};

Το OFPPC_PORT_DOWN bit δείχνει ότι η πόρτα είναι κάτω και δεν πρέπει να χρησιμοποιείται από το Openflow. Το OFPPC_NO_RECV bit δείχνει ότι τα πακέτα που λαμβάνονται σε αυτήν την πόρτα πρέπει να αγνοούνται. Το OFPPC_NO_FWD bit δείχνει ότι δεν πρέπει να στέλνονται πακέτα σε αυτήν την πόρτα. Το OFPPFL_NO_PACKET_IN bit δείχνει ότι το πακέτο παράγει μια αστοχία στον πίνακα και ότι δεν πρέπει να φύγει μήνυμα packet-in προς τον controller.

Γενικά, τα bit ρυθμίσεων τοποθετούνται από τον controller και δεν αλλάζουν από το switch. Αυτά τα bit μπορεί να είναι χρήσιμα από τον controller για να εφαρμόσει πρωτόκολλα όπως το STP ή BFD. Αν τα bits ρυθμίσεων της πόρτας αλλάξουν από το switch μέσω κάποιας άλλης διαχειριστικής διεπαφής, το switch στέλνει ένα OFPT_PORT_STATUS μήνυμα, ώστε να ενημερώσει τον controller για την αλλαγή.

Το πεδίο state περιγράφει την εσωτερική κατάσταση της πόρτας, και έχει την παρακάτω δομή

/* Current state of the physical port. These are not configurable from * the controller. */

enum ofp_port_state {

OFPPS_LINK_DOWN = 1 << 0, /* No physical link present. */

OFPPS_BLOCKED = 1 << 1, /* Port is blocked */

OFPPS_LIVE = 1 << 2, /* Live for Fast Failover Group. */

};

Τα port state bits δείχνουν την κατάσταση της φυσικής ζεύξης ή των πρωτοκόλλων του switch εκτός του openflow. Το OFPPS_LINK_DOWN bit δείχνει ότι η φυσική ζεύξη είναι πεσμένη. Το OFPPS_BLOCKED bit δείχνει ότι τα πρωτόκολλα εκτός openflow, όπως το 802.1D Spanning Tree, αποτρέπουν την χρήση αυτής της πόρτας.

Όλα τα bit της ofp_port_state είναι read-only και δεν μπορούν να αλλαχθούν από τον controller. Όταν ένα flag μιας πόρτας αλλάξει, το switch στέλνει ένα OFPT_PORT_STATUS μήνυμα στον controller για να τον ενημερώσει για την αλλαγή.

Ένα OpenFlow switch παρέχει περιορισμένη υποστήριξη για Quality of Service (QoS) μέσω ενός απλού μηχανισμού queuing. Μια ή περισσότερες ουρές μπορούν να προσδεθούν σε μια πόρτα και να χρησιμοποιηθούν για να αποδώσουν flow entries(εισόδους ροής) στο switch. Οι flow entries που αποδίδονται σε μια συγκεκριμένη ουρά θα αντιμετωπίζονται σύμφωνα με το configuration της ουράς. Κάθε ουρά έχει ένα σύνολο από ιδιότητες , κάθε μια ενός συγκεκριμένου τύπου και configuration.

enum ofp_queue_properties {

OFPQT_MIN_RATE = 1, /* Minimum datarate guaranteed. */

OFPQT_MAX_RATE = 2, /* Maximum datarate. */

OFPQT_EXPERIMENTER = 0xffff /* Experimenter defined property. */

};

2.5.2 Μορφή Επικεφαλίδας ταιριάσματος ροής

Η μορφή που χρησιμοποιείται για την επικεφαλίδα του πεδίου του ταιριάσματος ροής είναι η OpenFlow Extensible Match (OXM), η οποία είναι μια συμπαγής Type Length Value (TLV)μορφή. Κάθε OXM TLV έχει μήκος από 5 έως 259 bytes.

Τα πεδία της επικεφαλίδας του OXM TLV φαίνονται παρακάτω. Το μέγεθος της επικεφαλίδας είναι 32 bit

Name		Width	Usage
oxm_type	oxm_class	16	Match class: member class or reserved class
	oxm_field	7	Match field within the class
	oxm_hasmask	1	Set if OXM include a bitmask in payload
	oxm_length	8	Length of OXM payload

Εικόνα 2. 5: ***Πίνακας πεδίων επικεφαλίδας OXM TLV***

Η oxm_class είναι μια OXM κλάση ταιριάσματος που περιέχει τους σχετικούς τύπους ταιριάσματος. Το πεδίο oxm_field μια τιμή που σχετίζεται με την κλάση, εντοπίζοντας έναν από τους τύπους ταιριάσματος μέσα στην κλάση ταιριάσματος. Ο συνδυασμός της oxm_class και του oxm_field είναι συγκεντρωτικά το oxm_type. Το oxm_type συνήθως χαρακτηρίζει το πεδίο της επικεφαλίδας στο πρωτόκολλο, όπως ο τύπος Ethernet, αλλά μπορεί επίσης να αναφέρεται στα μεταδεδομένα του πακέτου, όπως την πόρτα του switch στην οποία ένα πακέτο έφτασε.

Το πεδίο oxm_hasmask καθορίζει εάν ένα OXM TLV περιέχει bitmask

Το πεδίο oxm_length είναι ένας θετικός ακέραιος που περιγράφει το μήκος του φορτίου του OXM TLV σε bytes. Το μήκος του OXM TLV, συμπεριλαμβανομένου και του header, είναι ακριβώς 4 + oxm_length bytes.

Για μια δοσμένη oxm_class, oxm_field, και oxm_hasmask τιμή, το oxm_length είναι σταθερό. Συμπεριλαμβάνεται μόνο για να επιτρέψει στο λογισμικό να κάνει parse τα OXM TLVs άγνωστου τύπου. (Παρόμοια, για μια δοσμένη oxm_class, oxm_field, και oxm_length τιμή, oxm_hasmask είναι σταθερό.)

2.5.3 Flow Matching

Το OXM TLV τοποθετεί περιορισμούς στα πακέτα που μπορούν να θεωρηθούν OpenFlow

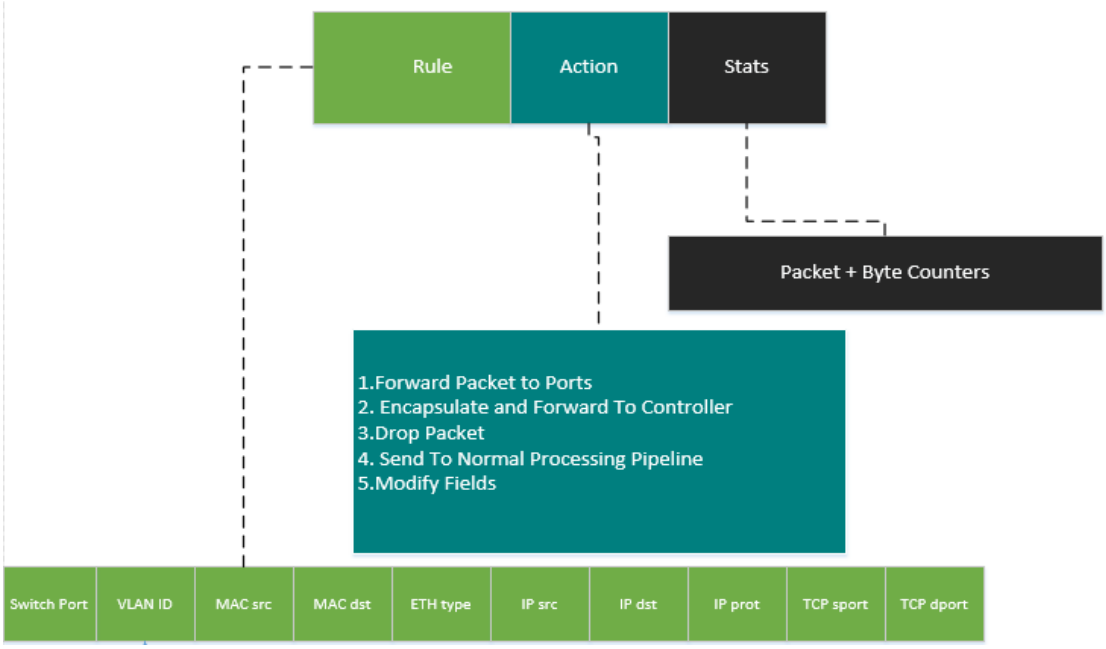
Εάν το oxm_hasmask είναι 0, τότε το σώμα του OXM TLV περιέχει μια τιμή για αυτό το πεδίο, που ονομάζεται oxm_value. Το OXM TLV κάνει το ταίριασμα μόνο για πακέτα που το αντίστοιχο πεδίο τους έχει τιμή ίση με oxm_value.

Εάν το oxm_hasmask είναι 1, τότε το σώμα του oxm_entry's περιέχει μια τιμή για το πεδίο (oxm_value), ακολουθούμενο από μια bitmask του ιδίου μήκους με την τιμή, που ονομάζεται oxm_mask. Το OXM TLV ταιριάζει μόνο πακέτα των οποίων η τιμή του αντίστοιχου πεδίου είναι ίση με την τιμή oxm_value για τα bits που καθορίζονται από την oxm_mask.

Όταν υπάρχουν πολλαπλά OXM TLVs, όλοι οι περιορισμοί πρέπει να τηρούνται: το πεδίο των πακέτων πρέπει να ταιριάζει όλα τα μέρη του OXM TLVs του OpenFlow.

2.6 OpenFlow Switch

Ένας OpenFlow switch θεωρείται μια συλλογή από flow tables που συμπεριλαμβάνουν τρείς στήλες: κανόνες, ενέργειες, και μετρητές. Η στήλη κανόνων προσδιορίζει το πεδίο επικεφαλίδας που καθορίζει την ροή. Οι κανόνες ταιριάζονται με την επικεφαλίδα των εισερχόμενων πακέτων. Εάν κάποιος κανόνας ταιριάζει, η ενέργεια από την στήλη ενεργειών εφαρμόζεται στο πακέτο και οι μετρητές από την στήλη των counters ανανεώνεται. Εάν ένα πακέτο ταιριάζει σε πολλαπλούς κανόνες, ο κανόνας με την υψηλότερη προτεραιότητα εφαρμόζεται. Κάθε κανόνας προσδιορίζει ένα ακριβές ταίριασμα με πεδίο header ή wild card π.χ. ANY. Το σύνολο από πιθανές ενέργειες είναι: προώθηση του πακέτου σε μια πόρτα εξόδου, τροποποίηση του πακέτου με κάποιον τρόπο, ή αποστολή του πακέτου σε επόμενο πίνακα. Ένας OpenFlow switch υποστηρίζει τρείς τύπους από OpenFlow πόρτες: φυσικές, λογικές και κατοχυρωμένες [24].



Εικόνα 2. 6: ***Πίνακας ροών, πηγή[16]***

Οι OpenFlow φυσικές πόρτες αντιστοιχούν σε διεπαφές του switch (π.χ. σε ένα Ethernet switch). Οι OpenFlow λογικές πόρτες είναι καθορισμένες από το switch πόρτες που δεν αντιστοιχούν σε διεπαφές υλικού του switch (π.χ. ενθυλάκωση πακέτων). Η μόνη διαφορά μεταξύ τν φυσικών πορτών με τις λογικές πόρτες είναι ότι ένα πακέτο που συνδέεται με μια λογική πόρτα μπορεί να έχει επιπλέον πεδίο metadata που ονομάζεται Tunnel-ID. Οι κατοχυρωμένες πόρτες καθορίζονται στα OpenFlow switch specification [24]. Ένα switch δεν απαιτείται να υποστηρίζει όλες τούς κατοχυρωμένες πόρτες, αλλά μόνο όσες καθορίζονται ως “required”

Κεφάλαιο 3°

3.1 Μειονεκτήματα της 3GPP Αρχιτεκτονικής

Η GTP-C διεπαφή καθορίζεται για σηματοδοσία μεταξύ πολλών στοιχείων του core network (MME-SGW/PGW) και υπηρεσιών. Μια αναποτελεσματική διαχείριση της διεπαφής θα είχε τα εξής αποτελέσματα: το χάσιμο της PDN σύνδεσης (IMS, Internet) και των αντίστοιχων υπηρεσιών, την απώλεια της ικανότητας εγκαθίδρυσης και ελευθέρωσης network bearers (GBR bearers για Voice πάνω από LTE (VoLTE)), τη απώλεια της ικανότητας να αναφέρει στο PGW / PCRF αλλαγές πληροφοριών για τον χρήστη (πληροφορίες τοποθεσίας για υπηρεσίες εκτάκτου ανάγκης και lawful intercept, αλλαγές στο Radio Access Transfer (RAT) ή QoS), λάθη χρέωσης και απώλεια κερδών. Η διεπαφή GTP-C πρέπει να βελτιώνεται συνεχώς για να υποστηρίζει προστασία από υπερφόρτωση και δυναμικό έλεγχο φορτίου καθώς προς το παρόν έχει περιορισμένη υποστήριξη σε αυτούς τους τομείς. Για τον λόγο αυτό, μηχανισμοί εντοπισμού και περιορισμού της υπερφόρτωσης στο επίπεδο του ελέγχου GTP-C πρέπει να εξερευνηθούν [21]. Οι τυποποιημένοι μηχανισμοί GTP-C load balancing βασίζονται στο ότι το MME πρέπει να χρησιμοποιεί Domain Name System (DNS) βάρη που είναι ημιστατικά "και τυπικά τοποθετούνται σύμφωνα με την χωρητικότητα του gateway κόμβου σε σχέση με τους άλλους gateway κόμβους" [21]. Σε αυτήν την περίπτωση τα DNS βάρη που σχετίζονται με αυτόν τον κόμβο γίνονται η λάθος πληροφορία. Όταν το gateway που απέτυχε επανεκκινεί, η τοποθέτηση της κίνησης, που βασίζεται σε DNS βάρη, διανέμει το ίδιο ποσό κίνησης σε διαφόρους gateways, ακόμα και αν το gateway που επανεκκινει δεν είναι φορτωμένο καθόλου, ενώ τα άλλα gateways υποστηρίζουν μεγαλύτερο φορτίο από ότι συνήθως. Για παράδειγμα, στην περίπτωση του SGW,ο παράγοντας του βάρους τοποθετείται ανάλογα με την χωρητικότητα του SGW κόμβου σε σχέση με άλλους SGW κόμβους που εξυπηρετούν την ίδια TA (Tracking Area). Το ίδιο ισχύει και κατά την διάρκεια επέκτασης του δικτύου(π.χ. όταν προσθέτουμε καινούργια SGW σε ένα συγκρότημα) [21].

Ο μηχανισμός του load-balancing είναι ανεπαρκής όταν ένας SGW έχει μερική αποτυχία (π.χ. ένα από τα στοιχεία του έχει αποτύχει), ή κατά την διάρκεια συγκεκριμένων διαδικασιών διατήρησης, μπορεί ακόμα να δουλεύει αλλά όχι κάτω από πλήρη χωρητικότητα. Από την φύση του, το πρωτόκολλο GTP-C δεν έχει σχεδιαστεί να αντιμετωπίζει ξαφνική αύξηση της συμφόρησης τόσο στο επίπεδο ελέγχου όσο και του χρήστη. Οι λύσεις που προτείνονται περιλαμβάνουν configuration νέων standard διεπαφών και αφήνουν να εννοηθούν τροποποιήσεις στην υποδομή του core network, η οποία είναι πολύ ευαίσθητη περιοχή για τους παρόχους.

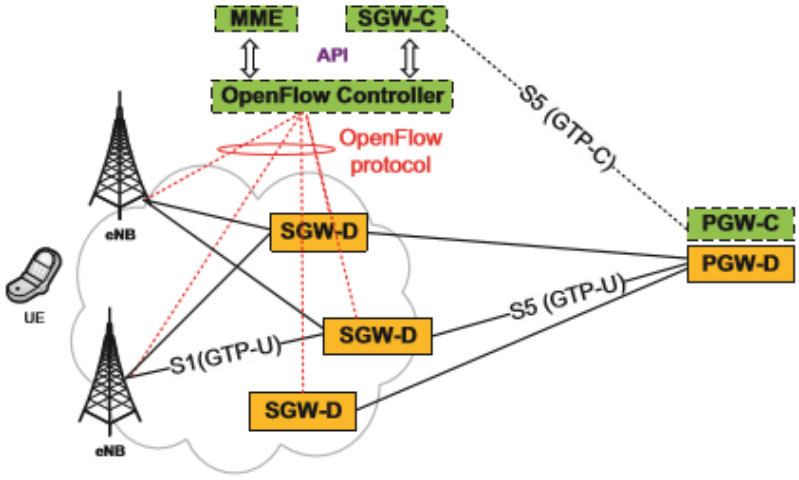
Ένα επιπλέον σημαντικό μειονέκτημα είναι η εγκαθίδρυση τούνελ (μεταξύ eNB, SGW και PGW) κατά την διαδικασία του Initial Attachment, ακόμα και αν δεν υπάρχουν δεδομένα να σταλούν. Οι TEID τιμές τοποθετούνται τοπικά σε κάθε κόμβο. Για τον λόγο αυτό, νέες τιμές TEID πρέπει να ανταλλάσσονται σε κάθε μετεγκατάσταση του κόμβου το οποίο επηρεάζει δραστικά την ελαστικότητα του δικτύου. Μέχρι τώρα δεν υπάρχει προκαθορισμένη πολιτική διαχείρισης ανά συνδρομητή εξαιτίας της υψηλής πολυπλοκότητας και η πρώτη φάση αφορά πολιτική συνολικού επιπέδου μαζί με μηχανισμούς

συμφόρησης. Τα 3GPP standards εξηγούν πως να χτίσουμε μονοπάτια μετάδοσης μεταξύ του UE και του PGW με καλά καθορισμένη QoS. Μέχρι τώρα ,το 3GPP έχει καθορίσει ένα εκτενές “bearer model” που εφαρμόζει τους μηχανισμούς διασφάλησης του QoS [13].

Κοιτώντας πέρα από τις προτεινόμενες λύσεις του 3GPP, το NFV και το SDN μπορούν να παρέχουν μακροπρόθεσμες λύσεις για το static load-balancing και τις προκλήσεις των μηχανισμών QoS . Για τη δημιουργία μιας πλήρους συμβατής λύσεις είναι σημαντικό να κατανοήσουμε την εξέλιξη και τις ικανότητες αυτών των τεχνολογιών, όπως επίσης και την σχέση: Cloud Computing / NFV,ή την σχέση μεταξύ NFV και SDN.

3.2 SDN και EPC

Σύμφωνα με το [19] τα κυψελωτά δίκτυα δεδομένων, όπως το LTE έχουν έλλειψη ελέγχου και πλήρης οπτικής του δικτύου, το οποίο δεν επιτρέπει τις on-demand υπηρεσίες συνδεσιμότητας. Το SDN βοηθά στο να ξεπεράσουμε αυτά τα μειονεκτήματα. Προτείνεται μια LTE αρχιτεκτονική βασισμένη στο OpenFow. Τα κύρια χαρακτηριστικά που εξετάζει το paper που προαναφέρθηκε είναι η ανθεκτικότητα (resilience) και το load balancing. Προτείνεται ένα καινούργιο επίπεδο ελέγχου στην αρχιτεκτονική LTE. Στην πρόταση αυτή, αντικαθιστώνται τα πρωτόκολλα ελέγχου που τρέχουν στις διαπαφές S1 MME (μεταξύ MME και eNB) και το S11 (μεταξύ MME και SGW) από το Openflow πρωτόκολλο, όπως δείχνει η παρακάτω εικόνα



Εικόνα 3. 1: αντικατάσταση των πρωτόκολλων ελέγχου που τρέχουν στις διαπαφές S1 MME (μεταξύ MME και eNB) και του S11 (μεταξύ MME και SGW) από το Openflow πρωτόκολλο

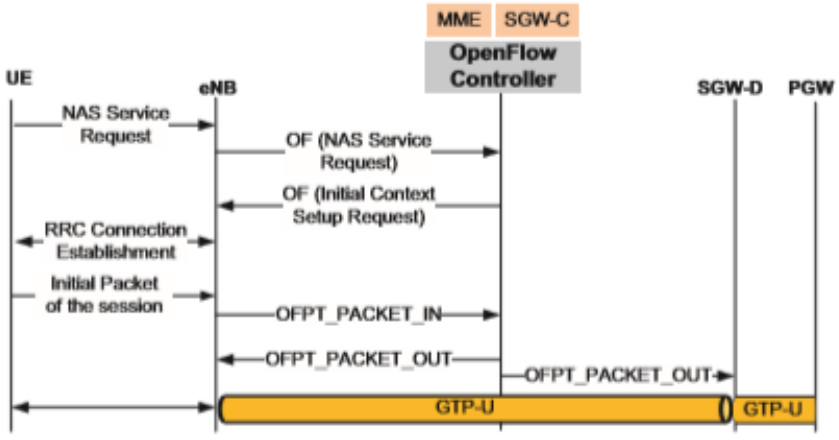
3.2.1 Αρχιτεκτονική LTE βασισμένη στο Openflow

Υπάρχει διαχωρισμός των λειτουργιών ελέγχου από τις λειτουργίες προώθησης δεδομένων στα SGWs. Σαν αποτέλεσμα, ολόκληρη η ευφυΐα στα SGW (SGW-C λογισμικό) και στο MME κεντρικοποιήται και τρέχει πάνω σε OpenFlow Controller (OF-ctr) σαν εφαρμογή. Η λειτουργία προώθησης δεδομένων πραγματοποιείται από το SGW επίπεδο δεδομένων (SGW-D). Η αρχιτεκτονική αποτελείται από τις παρακάτω οντότητες:

- OpenFlow Controller (OF-ctr): είναι το κύριο στοιχείο της αρχιτεκτονικής που διαχειρίζεται το επίπεδο προώθησης του eNB και SGW-D. Ο OF-ctr είναι υπεύθυνος για την εγκαθίδρυση των session του χρήστη και την παρακολούθηση του φορτίου στο επίπεδο δεδομένων.
- MME: είναι υπεύθυνο για την πιστοποίηση και επιβεβαίωση των UE, και για intra-3GPP διαχείριση κινητικότητας. Στην αρχιτεκτονική μας, το MME δεν είναι υπεύθυνο για την επιλογή του SGW και του PGW. Το MME επικοινωνεί με τον OF-ctr χρησιμοποιώντας Application Programming Interface (API).Η 3GPP διεπαφή μεταξύ του MME και του HSS διατηρείται.
- Το επίπεδο ελέγχου του SGW (SGW-C): αντιπροσωπεύει το κομμάτι ευφυΐας του SGW.είναι υπεύθυνο για την εγκαθίδρυση του GTP τούνελ και την τοποθέτηση των TEIDs. το SGW-C τοποθετεί μοναδικές τιμές TEID ανά session για την uplink κίνηση μέσα στη S1-U διεπαφή. Τοποθετεί επίσης μοναδικές τιμές TEID για την downlink κίνηση μέσα στην S5-U διεπαφή. Με το openflow πρωτόκολλο, ο Of-ctr μπορεί να τοποθετήσει μετρητές στο SGW-Ds για να παίρνει περιοδικά στατιστικά του φορτίου της κίνησης. Συγκρίνοντας το λαμβανόμενο φορτίο κίνησης μέσω των στατιστικών με τις δυνατότητες του SGW-D, ο OF-ctr μπορεί εύκολα να πάρει την κατάσταση του φορτίου κάθε SGW-D και να κάνει αποτελεσματικότερο load balancing (π.χ. βασιζόμενο στο τωρινό φορτίο SGW-Ds).
- Το SGW επίπεδο δεδομένων (SGW-D): αντιπροσωπεύει ένα προηγμένο OF switch που είναι ικανό να ενθυλακώνει/αποθυλακώνει GTP πακέτα. Αυτό το switch εφαρμόζει τους κανόνες που λαμβάνει από τον OF-ctr. Είναι υπεύθυνο για προώθησης πακέτων μεταξύ του eNB και του PGW.
- eNB: κρατάει τις ίδιες λειτουργίες που καθορίζονται από τα 3GPP standards. Είναι ικανό να χρησιμοποιεί το OF πρωτόκολλο για την προώθηση δεδομένων μέσω της S1 διεπαφής. Για το λόγο αυτό η προώθηση δεδομένων βασίζεται σε εντολές που λαμβάνονται από τον OF-ctr
- PGW: ακόμα έχει τις ίδιες λειτουργίες που καθορίζονται από τα 3GPP standards. Η τοποθέτηση των TEID τιμών στο SGW-C καθορίζονται μια φορά ανά session. Αυτές οι τιμές παραμένουν αμετάβλητες κατά την διάρκεια της μετακίνησης από το ένα SGW-D στο άλλο. Για την ακρίβεια, όταν υπάρχουν SGW-C εντολές του OF-ctr για μετεγκατάσταση του SGW-D για συγκεκριμένο session, ο OF-ctr απλά θα ενημερώσει στην eNB flow entry που συνδέεται με αυτό το session την IP διεύθυνση του και- νούργιου SGW-D. Επίσης, το SGW-C ενημερώνει την SGW-D IP διεύθυνση στο PGW.

3.2.2 Διαδικασία εγκαθίδρυσης επιπέδου δεδομένων

Αυτή η διαδικασία είναι απαραίτητη για κάθε καινούργιο session. Πρώτα, το UE στέλνει στο MME ένα NAS Service Request μήνυμα για να πάρει την εξουσιοδότηση για την εγκαθίδρυση του κομιστή των ασύρματων (radio) δεδομένων. Το UE στέλνει στο eNB το αρχικό πακέτο μέσω του εγκαθιδρυμένου κομιστή. Μετά, το eNB ελέγχει τους πίνακες ροών του. Καθώς δεν υπάρχει flow entry για το αρχικό πακέτο, το eNB στέλνει στον OF-ctr την επικεφαλίδα του πακέτου μέσω ενός OFPT_PACKET_IN μηνύματος. Επίσης, το eNB περιλαμβάνει σε αυτό το μήνυμα την τιμή του eNB-TEID για την downlink κίνηση στην S1 διεπαφή. Ο OF-ctr αναλύει την επικεφαλίδα του πακέτου για να αναγνωρίσει την IP διεύθυνση πηγής, την IP διεύθυνση προορισμού και τον τύπο του session. Ο OF-ctr παρουσιάζει αυτές τις πληροφορίες στο SGW-C. Βασισμένο στις IP διευθύνσεις και στα στατιστικά φορτίου που συγκεντρώνονται από τον OF-ctr, το SGW-C επιλέγει τον κατάλληλο SGW-D. Ο τύπος του session επιτρέπει στο SGW-C να αποφασίσει το κατάλληλο QoS. Για παράδειγμα, αν το πακέτο ανήκει στο VoIP και το ήδη επιλεγμένο SGW-D είναι υπερφορτωμένο, ο OF-ctr αποφασίζει να τοποθετήσει το πακέτο σε ένα άλλο SGW-D με μικρότερο φορτίο. Το SGW-C στέλνει πίσω στο OF-ctr την SGW-D IP διεύθυνση, τις τιμές SGW-TEID και το επίπεδο του QoS. Ο OF-ctr δημιουργεί μια flow entry για το πακέτο που σχετίζεται με το session και την στέλνει στο eNB μέσω του μηνύματος OFPT_PACKET_OUT. Το πεδίο της ενέργειας αυτού του flow entry περιλαμβάνει την SGW-D IP διεύθυνση και την τιμή SGW-TEID για την uplink κίνηση στην S1 διεπαφή. Παρόμοια, ο OF-ctr δημιουργεί και στέλνει στον SGW-D μια flow entry σχετική με αυτό το session μέσω του OFPT_PACKET_OUT μηνύματος. Το πεδίο ενέργειας της flow entry περιλαμβάνει την eNB IP διεύθυνση, την eNB-TEID τιμή, την SGW-TEID τιμή για την uplink κίνηση στην S1 διεπαφή, την PGW IP διεύθυνση, την τιμή PGW-TEID, και την τιμή SGW-TEID για την downlink κίνηση στην S5 διεπαφή. Ο OF-ctr αποφασίζει την τιμή του χρόνου αεργίας του UE και την συμπεριλαμβάνει σε κάθε flow entry. Για τον λόγο αυτό, όταν η τιμή του χρόνου αεργίας του UE σε μια flow entry λήγει και κανένα πακέτο δεν καταφθάνει, το eNB ή το SGW-D απλά διαγράφει αυτό το flow entry. Σε αντίθεση με την σημερινή LTE/EPC αρχιτεκτονική, καμία επιπλέον σηματοδότηση δεν χρειάζεται για να ελευθερώσουμε τον κομιστή πρόσβασης(access bearer).



Εικόνα 3. 2: *Παρουσιάζεται η εγκαθίδρυση του επιπέδου δεδομένων*

Αντοχή στις αποτυχίες βασισμένη στην αρχιτεκτονική OF LTE : Η αποτυχία του SGW μπορεί εύκολα να διαχειριστεί. Καθώς ο OF-ctr και το SGW-Ds ανταλλάσουν περιοδικά Echo Request/Reply μηνύματα, ο OF-ctr μπορεί να εντοπίσει κάθε SGWD αποτυχία. Στον εντοπισμό της SGW-D1 αποτυχίας, το SGWC επιλέγει SGW-D2 για τα sessions που επηρεάστηκαν από την αποτυχία. Το SGW-C ενημερώνει την SGW-D IP διεύθυνση που βρίσκεται στο PGW μέσω ενός μηνύματος Modify Bearer Request. Προτείνεται η χρήση της ίδιας διεπαφής για την μεταβίβαση της επικοινωνίας μεταξύ του SGW-C και PGW. Όπως καθορίζεται στην αρχιτεκτονική, η τιμή του SGW-TEID για την κίνηση στο downlink στην S5 interface παραμένει η ίδια για τα sessions που επηρεάζονται. Μετά από αυτό, ο OF-ctr ενημερώνει την SGWD IP διεύθυνση στο eNB μέσω OFPT_MODIFY_STATE μηνύματος. Το πλεονέκτημα μιας κεντρικοποιημένης λειτουργίας τοποθέτησης TEID που σχετίζεται με τα SGWs. Το SGWC δεν δημιουργεί νέες τιμές TEID κατά την διάρκεια της διαδικασίας αποκατάστασης. Ο OF-ctr ενημερώνει τις flow entries στα eNBs με τις καινούργιες SGW-D IP διευθύνσεις. Καθώς τα eNBs τα ίδια για κάθε session, οι τιμές eNB-TEID για την downlink κίνηση στην S1 διεπαφή δεν αλλάζει. Ο OF-ctr τοποθετεί τις νέες flow entries στο SGW-D 2 μέσω ενός PACKET_OUT μηνύματος.

Load Balancing βασισμένο στην αρχιτεκτονική OF LTE: το να παίρνει περιοδικά στατιστικά για το φορτίο του SGW-D είναι ένα από τα πλεονεκτήματα της εφαρμογής του OpenFlow στο EPC. Αυτά τα στατιστικά είναι σημαντικά για πιο πετυχημένο load balancing. Για παράδειγμα, βασιζόμενοι σε στατιστικά πραγματικού χρόνου που παρουσιάζονται από τον OF-ctr και τον τύπο του session (π.χ. που καθορίζεται από την επικεφαλίδα του πακέτου), το SGW-C μπορεί να ισορροπήσει τη κίνηση στα SGW-Ds οδηγώντας σε καλύτερη διανομή της κίνησης. Σε αντίθεση με τα 3GPP standards, η αρχιτεκτονική που προτείνεται μπορεί προσωρινά να αποφορτίσει τα SGW μετακινώντας μερικά sessions σα άλλα SGW στο ίδιο domain.

3.2.3 Προκλήσεις εφαρμογής

Για την εφαρμογή της προτεινόμενης αρχιτεκτονικής, πολλές προκλήσεις πρέπει να ξεπεραστούν:

• Για λειτουργίες ελέγχου στο SGW, όπως η τοποθέτηση των TEID, πρέπει πρώτα να διαχωριστεί το επίπεδο ελέγχου από το επίπεδο προώθησης. Αυτές οι λειτουργίες ελέγχου πρέπει να τρέχουν σαν εφαρμογές πάνω στον OF controller. Παρόμοια, οι MME λειτουργίες πρέπει να μετατραπούν σε εφαρμογές που επίσης τρέχουν στο OF ελεγκτή. Παρομοίως, η SGW και PGW λειτουργίες επιλογής πρέπει να μετατοπιστεί σε άλλη εφαρμογή.

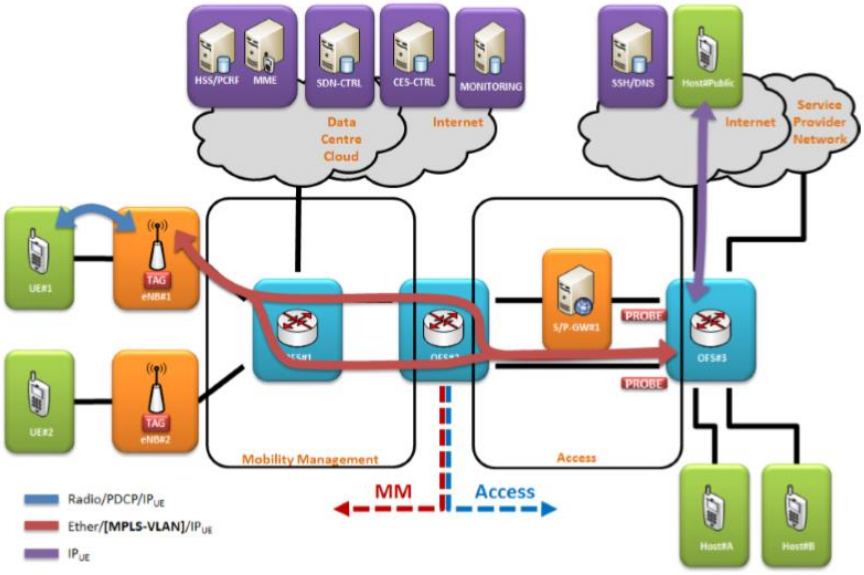
• Ο OF controller πρέπει να έχει ολική οπτική της τοπολογίας του domain και γνώση για τα χαρακτηριστικά των συσκευών του δικτύου. Αυτό απαιτείται από την SGW και την PGW επιλογή.

• Το SGW-C είναι πιθανό να χρειάζεται μεγαλύτερη βάση δεδομένων για να αποθηκεύσει πληροφορίες σχετικές με τις διαδικασίες δικτύωσης κάτω από τον τομέα ελέγχου (όπως τις ενεργές των flow entries, τις τιμές TEID, κ.τ.λ.). Κατά συνέπεια, η κατάλληλη μνήμη, IO, και CPU ικανότητες απαιτούνται για να αποθηκεύσουν τέτοιες πληροφορίες και να υπολογίσουμε την κατάλληλη διαχείριση κάθε session (π.χ., αποφάσεις για δρομολόγηση, κινητικότητα, και την συμπεριφορά στο QoS).

• Το OF πρέπει να επεκτείνεται στην μετατροπή των UE-MME ανταλλαγών με διαφανή τρόπο, π.χ. τις ανταλλαγές πιστοποίησης. Επιπλέον το OF switch πρέπει να επεκταθεί με τις λειτουργίες GTP ενθυλάκωσης/αποθυλάκωσης. Για παράδειγμα, η τωρινή δομή δεδομένων της πόρτας του OF switch δεν περιέχει τις GTP παραμέτρους, δηλαδή τις τιμές TEID προορισμού και πηγής.

3.3 Virtual EPC με λειτουργίες SDN στα Mobile Backhaul δίκτυα

Στο περιβάλλον όπου οι λειτουργίες του δικτύου ενσωματώνονται σε cloud servers σαν VNFs, κάθε στοιχείο του EPC δικτύου (πχ . MME , S/P-GW) θα τρέχει στο δικό του εικονικό μηχάνημα. Το γεγονός ότι το vEPC τρέχει σε διαφορετικό virtual machines επιτρέπει στους διαχειριστές να προσθέτουν στοιχεία δικτύου όταν είναι απαραίτητο ή να αυξήσουν τους πόρους των virtual machines για να διαχειριστούν το επιπλέον φορτίο. Το vEPC τρέχει σε virtual machines σαν λειτουργία VNF χρησιμοποιώντας τον OpenFlow controller σαν το επίπεδο virtualization για να διαχειριστεί τα OpenFlow φυσικά switches.

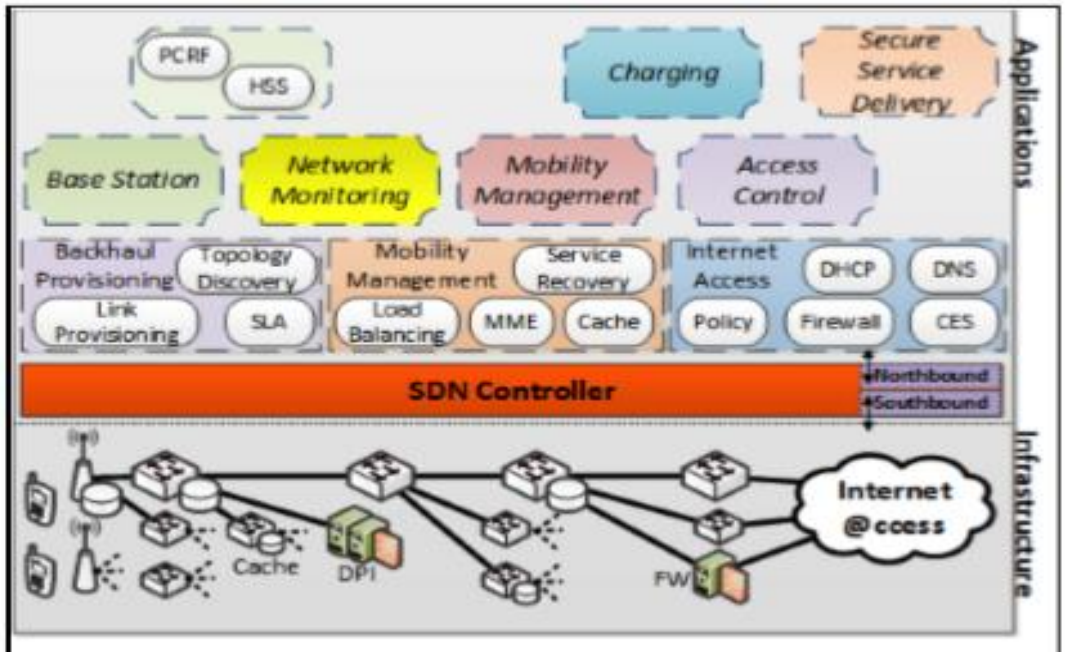


Εικόνα 3. 3: PoC σενάριο 3. SDN backhaul με gtp tunneling

Στο σενάριο που περιγράφεται [18] με την παραπάνω εικόνα μπορούμε να διακρίνουμε ότι η χρήση του SDN αντικαθιστά τελείως το επίπεδο δεδομένων των standard στοιχείων του δικτύου όπως το S/P-GW. Στοιχεία όπως το PCRF επίσης αντικαθιστώνται από εφαρμογές SDN που παρέχουν παρόμοια λειτουργία πάνω στον SDN OpenFlow controller. Σε αυτό το σενάριο επιπρόσθετα κουτιά virtualization μπορούν να προστεθούν για να παρέχουν λειτουργίες NFV για την διαχείριση συγκεκριμένων flows. Αυτά τα μεσαία κουτιά μπορούν να αντανακλούν πακέτα HTTP σε proxy servers για την καλύτερη προσωρινή αποθήκευση στην μνήμη ή μπορούν να αναγνωρίσουν ύποπτα flows και να τα ανακατευθύνουν σε firewalls ή honeypots. Σε αυτό το σενάριο ενσωματώνουμε το επίπεδο ελέγχου των S/P-GW με τον SDN OpenFlow controller. Οι λειτουργίες των S/P-GW περιορίζονται στο έλεγχο στο cloud. Αφαιρούμε εντελώς το GTP tunneling και χρησιμοποιούμε το eNB για να στείλουμε τα πακέτα δεδομένων σε μια καθορισμένη μορφή που σε αυτήν την περίπτωση υποστηρίζεται από το OpenFlow πρωτόκολλο. Το QoS μπορεί να καθορίζεται σε L2 tag QoS bits. Αυτό το σενάριο μπορεί επίσης να βοηθήσει στην προσωρινή αποθήκευση στο SDN καθώς έχουμε τα πακέτα δεδομένων του UE διαθέσιμα στο eNB και μπορούν να εκτρέπονται σε proxy servers. Το δίκτυο μπορεί να βελτιώνεται την ώρα που τρέχει βασιζόμενο σε πληροφορίες που συλλέγονται από ανιχνευτές κίνησης.

Στο [20] φαίνεται πως δομείται το σύνολο των λειτουργιών ελέγχου του LTE σε ένα group από SDN εφαρμογές για 5G, με τρόπο τέτοιο ώστε το επίπεδο δεδομένων των κινητών δικτύων να χτιστεί βασιζόμενο σε στάνταρ Openflow switches και Ethernet switches. Η 3GPP αρχιτεκτονική είναι βασισμένη σε συγκεκριμένες μεθόδους tunneling και το Openflow δεν υποστηρίζει τέτοιες μεθόδους.

Το 3GPP έχει κάνει βήματα για τον διαχωρισμό των επιπέδων ελέγχου και δεδομένων, καθώς και των αντίστοιχων στοιχείων τους.



Εικόνα 3. 4: Εμφανίζεται ο έλεγχος του δικτύου σαν ένα σύνολο από SDN εφαρμογές

Στην παραπάνω εικόνα εμφανίζεται ο έλεγχος του δικτύου σαν ένα σύνολο από SDN εφαρμογές. Αυτές οι εφαρμογές το δικτύου γίνονται orchestrated μέσω του Controller Northbound API, με τρόπο τέτοιο ώστε πολλαπλές SDN εφαρμογές να χρησιμοποιούνται χωρίς να έχουμε σύγκρουση μεταξύ τους.

3.3.1 Mobile Backhaul

Για να αφήσει τις SDN εφαρμογές να διαχειριστούν το δίκτυο τοποθετείται ένας OF switch (ονομάζεται eOFS) σαν το πρώτο στοιχείο με το οποίο τα enodeB είναι συνδεδεμένα. Το eOFS θα βάλει ετικέτα στα πακέτα από τον χρήστη ως το Internet. Μια ταιριαστή μορφή ενθυλάκωσης θα είναι η 802.1ah. Το δεύτερο OF switch (ονομάζεται mOFS) απαιτείται σαν σημείο εισόδου για την πρόσβαση στο Internet. Στο outbound interface του mOFS Mobility as a Service (MaaS) παρουσιάζεται στον πάροχο της πρόσβασης στο Internet. Για την κίνηση από πολλά eOFSs σε λίγα mOFS θα χρησιμοποιήσουμε έναν συνδυασμό από Carrier Grade (CG) Ethernet Switches και OF switches. Μπορούμε να απομονώσουμε κάθε κινητό στο υποδίκτυο του χρησιμοποιώντας το 802.1ah. Η κίνηση από τα eNBs στο σημείο εισόδου στο Internet και πίσω μπορεί να γίνει η μεταγωγή της μέσω του δικτύου που περιγράφηκε. Για σκοπούς load balancing κάθε ζεύξη στο Internet πρέπει να είναι προσβάσιμη από κάθε eNB μέσω πολλών μονοπατιών. Είναι ευνοϊκό που το MaaS διατηρεί το σημείο πρόσβασης στο Internet ενός κινητού σταθερό ενώ αυτό βρίσκεται σε κατάσταση roaming στο κινητό δίκτυο.

Μία από τις δουλειές της Mobile backhaul εφαρμογής είναι να σετάρει και να διαχειρίζεται την υπηρεσία δρομολόγησης στα CGE switches μεταξύ eOFS και mOFS και να φροντίζει για ανάκαμψη από λάθος στο δίκτυο. Στην περίπτωση χρήσης 802.1ah για ενθυλάκωση, η I-SID τιμή σημαδεύει το μονοπάτι μεταξύ

ενός eOFS και ενός iOFS. Η ετικέτα B-VLAN μπορεί να χρησιμοποιειθεί για να διαχωρίσει την κίνηση διαφορετικών εικονικών operators εάν είναι απαραίτητο και τέλος η C-VLAN αναγνωρίζει έναν χρήστη σε ένα eNB. Στην περίπτωση της MPLS ενθυλάκωσης, ένα label stack θα εξυπηρετούσε τον ίδιο σκοπό με ένα VLAN tag.

3.3.2 Mobility Management App

Όταν μια συσκευή κινείται από μία περιοχή ενός eNB σε μια άλλη περιοχή και σε άλλο eNB,ο κανόνας στο mOFS για τη συσκευή πρέπει να τροποποιηθεί και ένας καινούργιος κανόνας μπορεί να χρειάζεται να δημιουργηθεί για το καινούργιο eOFS. Εάν το καινούργιο eNB είναι κάτω από το ίδιο eOFS όπως το προηγούμενο, τότε είναι αρκετό να τροποποιήσουμε τον προηγούμενο κανόνα στο eOFS. Επίσης πρέπει να φροντίσουμε την ισορροπία του φορτίου μεταξύ των εναλλακτικών μονοπατιών μεταξύ ενός eNB και ενός mOFS. Η εφαρμογή Mobility Management επιλέγει το μονοπάτι για μια συσκευή. Για load balancing αποφάσεις χρειάζεται είσοδος από την εφαρμογή παρακολούθησης του δικτύου Monitoring App. Η Mobility Management (MM) εφαρμογή ενσωματώνει το MME. Επιπρόσθετα , πρέπει να διαχειρίζεται την ποιότητα υπηρεσιών για κάθε χρήστη, να ισορροπεί το φορτίο μεταξύ διαφορετικών μονοπατιών και να δρομολογεί τον χρήστη σε μια προσωρινή μνήμη, όταν αυτό είναι ικανό. Σε αντίθεση με τη σημερινή κατάσταση, σε αυτόν τον σχεδιασμό κανένα χαμηλότερο IP επίπεδο δεν χρησιμοποιείται για να μεταφέρει την κίνηση του χρήστη από και προς το Internet. Εκτός από ένα τούνελ δρομολόγησης στο Internet, προτείνουμε την χρήση τούνελ μεταγωγής. Για την διαχείριση της κινητικότητας, τα eNBs χρειάζεται να συνδεθούν με τους γείτονές τους. Για τον λόγο αυτό, ένας κατάλληλος τρόπος σύνδεσης των eNBs με την ιεραρχία του δικτύου είναι το 802.1ad. Στο πλαίσιο του 802.1ad, ένα VLAN tag αναγνωρίζει έναν χρήστη ενώ το άλλο αναγνωρίζει το “Internet VLAN” που οδηγεί στο eOFS όπου η κίνηση μεταγάγετε με 802.1ah, ή την εναλλακτική των VLANs η οποία μεταγάγετε είτε στα eOFS ή υψηλότερα σε γειτονικά eNB. Από την στιγμή που η δεπαφή από eNB σε eNB χρησιμοποιείται για διαχείριση κινητικότητας, προτείνουμε ότι η εφαρμογή MM App θα προβλέπει αυτά τα μονοπάτια μεταγωγής. Ενώ η προώθηση των πακέτων βασίζεται στο switching, κάθε eNB έχει IP routing λειτουργικότητα για eNB σε eNB επικοινωνία. Για τον σκοπό αυτό η εφαρμογή MM App θα τοποθετεί IP διευθύνσεις σε eNBs. Εναλλακτικά, το Ethernet routing μπορεί να χρησιμοποιηθεί.

3.3.3 Εφαρμογή Πρόσβασης

Ο ρόλος της εφαρμογής Access App είναι να τοποθετεί την IP διεύθυνση σε μια κινητή συσκευή. Αυτή η διεύθυνση μπορεί να είναι ιδιωτική. Για τον λόγο αυτό η εφαρμογή Access App παρέχει ένα σημείο πρόσδεσης στο Internet και στα δίκτυα παροχής υπηρεσιών σε κάθε κινητή συσκευή με το να ελέγχει το iOFS. Το σημείο πρόσδεσης πρέπει να είναι όσο πιο σταθερό γίνεται ενώ το κινητό κινείται και κάνει λειτουργίες roaming σε ξένα δίκτυα. Η εφαρμογή πρόσβασης Access App θα διαχειριστεί το downstream

load balancing και τα firewalls. Πιστεύουμε ότι όλες αποδοχές των flow πρέπει να διαχωρίζονται από μια πολιτική που είναι μέρος των πληροφοριών συνδρομής του χρήστη. Οι πολιτικές επίσης μπορούν να είναι δυναμικές, π.χ. διαχείριση απομακρυσμένων hosts διαφορετικά, βασιζόμενοι στην φήμη που παράγεται από ένα σύστημα έμπιστης διαχείρισης. Επιπλέον, προτείνουμε την χρήση συνεργατικών firewall που επιτρέπουν ερωτήσεις σε, π.χ. , στο firewall του αποστολέα και αρχές πιστοποίησης πριν πάρουν την τελική απόφαση αποδοχής. Αυτό επιτρέπει την επίλυση των ορίων μεταξύ κλειστών και ανοικτών δικτύων. Μια κινητή συσκευή κάτω από ένα συνεργατικό firewall είναι προσβάσιμη χρησιμοποιώντας το πλήρως αρμόδιο όνομα domain του host (FQDN), μια αποδεκτή ταυτότητα γι τον εντοπισμό του δρομολογητή iOFS. Το τούνελ παροχής υπηρεσιών και το mobile backhaul τούνελ δένονται μαζί στο iOFS από μια κατάσταση δέσμευσης που διαχειρίζεται από το Firewall. Το Realm Gateway, ένα στοιχείο της εφαρμογής πρόσβασης Access App μπορεί να αποδέχεται κίνηση απευθείας από το Internet .

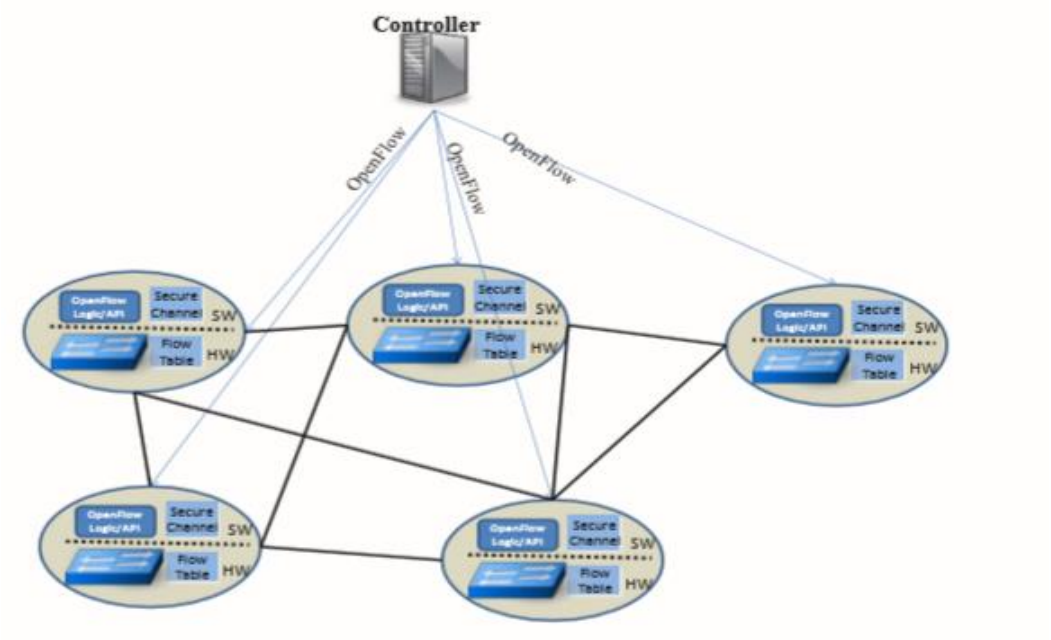
Η μετακίνηση από απλά firewalls δικτύου, που εφαρμόζουν τους ίδιους κανόνες σε κάθε πελάτη, σε συνεργατικά firewalls με κανόνες πρόσβασης χωριστούς για τον κάθε χρήστη, οι οποίοι διαχειρίζονται από την αρχιτεκτονική διαχείρισης πολιτικής του 3GPP στο 5G δικαιολογείται από την ανάγκη για μπλοκάρισμα όλων των πακέτων με πλαστογραφημένη διεύθυνση πηγής(source address spoofing), και όλα τα DDoS πακέτα από το να φτάσουν στο κινητό δίκτυο. Επίσης δικαιολογείται από την ανάγκη διαχείρισης της προσβασιμότητας κάθε συσκευής ανά εφαρμογή και ανά χρήστη δυσκίνητη NAT διάσχιση. Στην λύση που προτείνεται, είναι αρκετό για όλες τις κινητές συσκευές να έχουν μόνο μια private διεύθυνση. Για τον λόγο αυτό, η κλιμάκωση σε οποιοδήποτε αριθμό από χρήστες ή συσκευές στο 5G δεν εξαρτάται στην επιτυχία του IPv6.

3.3.4 Secure Service Delivery App

Παρουσιάζεται και η εφαρμογή Secure Service Delivery App. Με τον όρο δίκτυο παροχής υπηρεσιών εννοούμε το δίκτυο που συνδέει δύο κινητά δίκτυα ή ένα κινητό δίκτυο με έναν σταθερό πελάτη δικτύου ή με ένα απομακρυσμένο datacenter που έχει τις επιθυμητές εφαρμογές ή το επιθυμητό περιεχόμενο. Προτείνεται ότι με την εφαρμογή των εννοιών του SDN στην παροχή υπηρεσιών μπορούμε να βρούμε πλεονεκτήματα όπως η διασφάλιση της διεργασίας παροχής υπηρεσιών και να πάρουμε τα μέγιστα πλεονεκτήματα από την οικονομία με την χρήση φτηνών switches και γενικού hardware για διεργασίες ελέγχου. Ο ελάχιστος στόχος του δικτύου παροχής υπηρεσιών είναι να εξαφανιστεί το source address spoofing και οι DDoS επιθέσεις.

3.4 SDMN-Software Defined Mobility Networks

Είναι ευρέως αποδεκτό ότι τα δίκτυα του μέλλοντος θα απαιτούν μεγαλύτερο βαθμό επίγνωσης και καλύτερης χρήσης των πηγών του δικτύου. Όλα αυτά μπορεί να επιτευχθούν με τη βοήθεια του SDN. Το SDN αναμένεται να είναι το κλειδί στην ανάπτυξη της τηλεπικοινωνιακής υποδομής για τις αυξανόμενες ανάγκες των μελλοντικών κινητών δικτύων οδηγώντας στα Software Defined Mobile Network (SDMN). Το SDMN είναι μια προσέγγιση του δικτύου όπου το επίπεδο ελέγχου διαχωρίζεται από συγκεκριμένο υλικό που χρησιμοποιείται στις τηλεπικοινωνίες και παρέχεται σαν εφαρμογή λογισμικού. Το χαρακτηριστικό του SDMN μπορεί να είναι η απλοποίησης των routers και των switches με την μετακίνηση του CP σε έναν κεντρικοποιημένο server, τον ελεγκτή. Ο controller έχοντας τον συνολικό έλεγχο του δικτύου μπορεί να μειώσει την συμφόρηση με την χρήση εφαρμογών διαχείρισης κίνησης και να βελτιστοποιήσει την τοποθέτηση των πόρων του δικτύου. Στην επόμενη εικόνα φαίνεται η επικοινωνία μεταξύ του controller και των switches σε ένα SDN δίκτυο. Αυτή η επικοινωνία βασίζεται σε καλά καθορισμένα API όπως το OpenFlow. Το SDN μετατρέπει τις συσκευές του δικτύου όπως τα switches και τα routers σε προγραμματιζόμενες συσκευές. Επιπλέον, το SDN πρέπει να βελτιώνει την μεταφορά των δεδομένων μετά την απλοποίηση των λειτουργιών των switches. Η προτεινόμενη προσέγγιση είναι να εξορθολογήσουμε τις συσκευές του δικτύου έτι ώστε οι διεργασίες τους να βασίζονται στην προώθηση. Τα μονοπάτια θα πρέπει να υπολογίζονται από τον controller.



Εικόνα 3. 5**:** Δίκτυο βασισμένο στο SDN με τα switches να διαχειρίζονται από τον controller

Η SDN θα επιτρέψει την ενεργοποίηση ενός συνόλου νέων σεναρίων χρήσης:

- Διαχωρισμό των ατομικών ροών κίνησης ώστε να μοιράζονται διαθέσιμες πηγές που υποστηρίζουν οι Mobile Virtual Network Operator (MVNO).

- Βέλτιστη επανακατεύθυνση των ροών σε συγκεκριμένες εφαρμογές ή υπηρεσίες.

- Αποτελεσματική διαχείριση και χρήση των πηγών (π.χ. βελτίωση της χρήσης ενεργείας).

Παρόλα αυτά, το SDMN έχει συγκεκριμένους περιορισμούς όπως η αποτελεσματική διαχείριση της κινητικότητας και το scalability για τον χειρισμό μεγάλου όγκου ροών χρηστών κάτω από κάθε radio access δίκτυο. Το Software Defined Networking καθορίζεται για τα σταθερά δίκτυα άλλα για τα κινητά δίκτυα υπάρχουν διαφορετικές απαιτήσεις. Τα SDMN είναι μια επέκταση του SDN, που ενσωματώνει λειτουργίες κινητικότητας. Παραδείγματα λειτουργιών κινητικότητας είναι το mobility management, η αποτελεσματική προστασία της διεπαφής αέρα από ανεπιθύμητη κίνηση και η συχνή χρήσης του tunneling στην μεταφορά πακέτων. Η έκδοση του OpenFlow βασίζεται σε ενεργούς κανόνες που ο controller τοποθετεί στα switch όταν ένα πακέτο χωρίς τους απαραίτητους διαχειριστικούς κανόνες λαμβάνεται. Κατάλληλη ενσωμάτωση του SDN στην LTE αρχιτεκτονική απαιτείται για να διασφαλίσουμε ότι κεντρικοποιημένοι controllers μπορούν να διατηρήσουν την κατάσταση του δικτύου (π.χ. πληροφορίες τοπολογίας) και να αντιδράσουν σε αλλαγές τοπολογίας. Το SDN χρειάζεται να διαχειρίζεται όλα τα γεγονότα κινητικότητας που προέρχονται από χρήστες στα δίκτυα πρόσβασης χωρίς την συμφόρηση των καναλιών ελέγχου μεταξύ controllers και switching κόμβων. Η συμφόρηση μπορεί να συμβεί αν πολλά πακέτα δεδομένων πρέπει να σταλούν στον controller για την δημιουργία ενός κανόνα.

3.4.1 SOFTWARE DEFINED NETWORKS ΕΝΣΩΜΑΤΩΜΕΝΑ ΣΤΟ LTE

Η ενσωμάτωση του SDN στα κινητά δίκτυα, ώστε να γίνουν SDMN απαιτούν αρκετές αρχιτεκτονικές σχεδίασης:

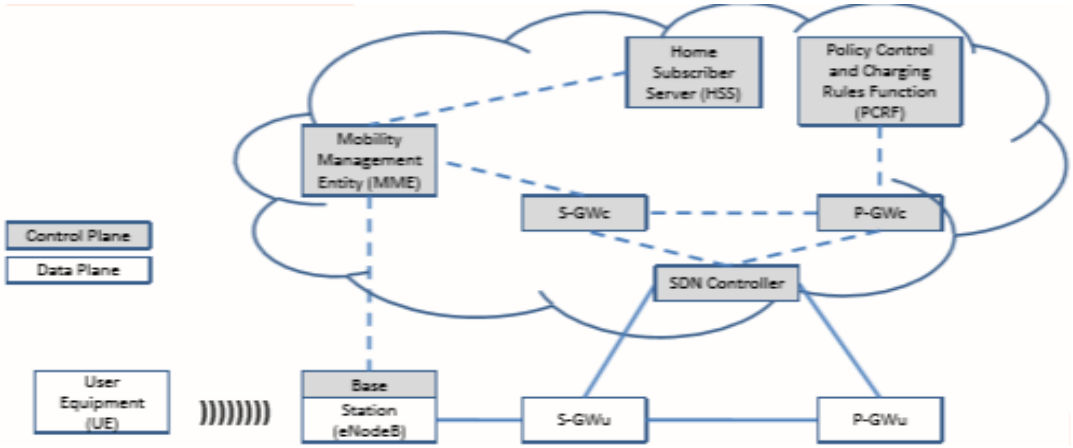
- Η τοποθεσία του SDMN controller: μπορεί να είναι μέρος του MME για να είμαστε ενήμεροι για απαιτήσεις κινητικότητας, ή μπορεί να τοποθετείται σαν μέρος του S/PGW για τον έλεγχο του δικτύου μεταφοράς.

- Η διανομή του controller: Μπορεί να είναι ένας απλός controller ή πολλοί controllers διανεμημένοι κοντά στο access network αλλά διατηρώντας ιεραρχική τοπολογία μεταξύ των controllers στα access δίκτυα και εντατικοποιημένη στο core δίκτυο.

Η ενσωμάτωση του SDN με το LTE δίκτυο είτε ως μέρος του MME ή ως μέρος του S/P-GW πρέπει να ακολουθεί εξελιχτηκές διεργασίες. Ο αντικειμενικός σκοπός είναι να διατηρήσουμε τα τωρινά IP δίκτυα και να συμπεριλάβουμε τα SDMN για να βελτιωθούν τα LTE δίκτυα.

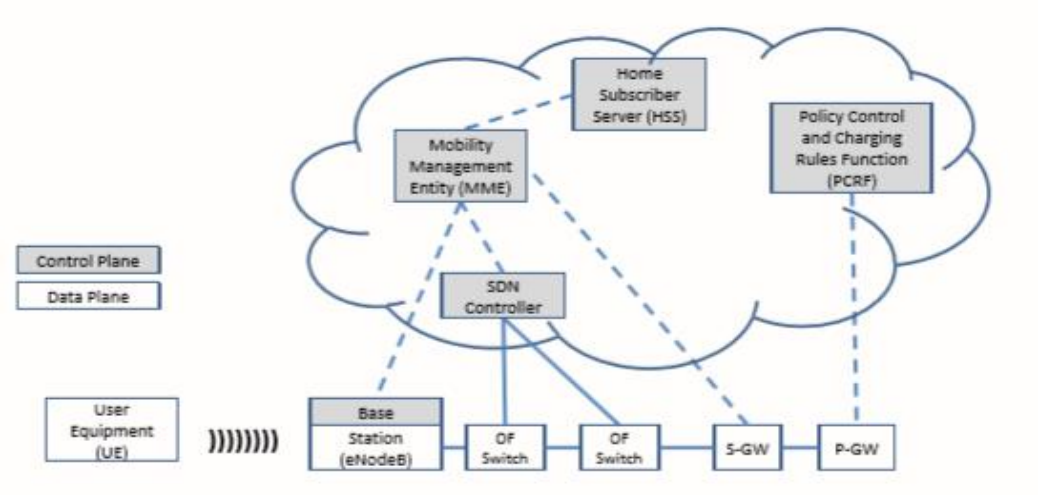
Στην παρακάτω εικόνα φαίνεται η ενσωμάτωση του SDN στην LTE αρχιτεκτονική. Αυτή η επιλογή αποτελείτε από τον διαχωρισμό του S/P-GW σε επίπεδο λογικό και δεδομένων. Το λογικό μέρος του S/P-GW (π.χ. S/PGWc) παρέχει τη τοποθέτηση των IP διευθύνσεων για το UE και την εφαρμογή του TFT στις ροές δεδομένων του χρήστη. Το επίπεδο δεδομένων του S/P-GW (π.χ. S/P-GWu) παρέχει το GTP tunneling τερματισμό και την διατήρηση των GTP tunnels κατά την διαδικασία του handover. Το λογικό μέρος του S/P-GW ενσωματώνεται με τον SDN controller για την διαχείριση του TFT στο S/P-GWu. Τα

υπόλοιπα στοιχεία του δικτύου διατηρούνται χωρίς καμία αλλαγή και το MME επικοινωνεί με το S/P-GWc.



Εικόνα 3. 6: *Ενσωμάτωση του SDN στην LTE αρχιτεκτονική*

Η δεύτερη επιλογή για την ενσωμάτωση του SDN στην LTE αρχιτεκτονική αποτελείται από την ζεύξη του SDN controller με το MME όπως φαίνεται παρακάτω.

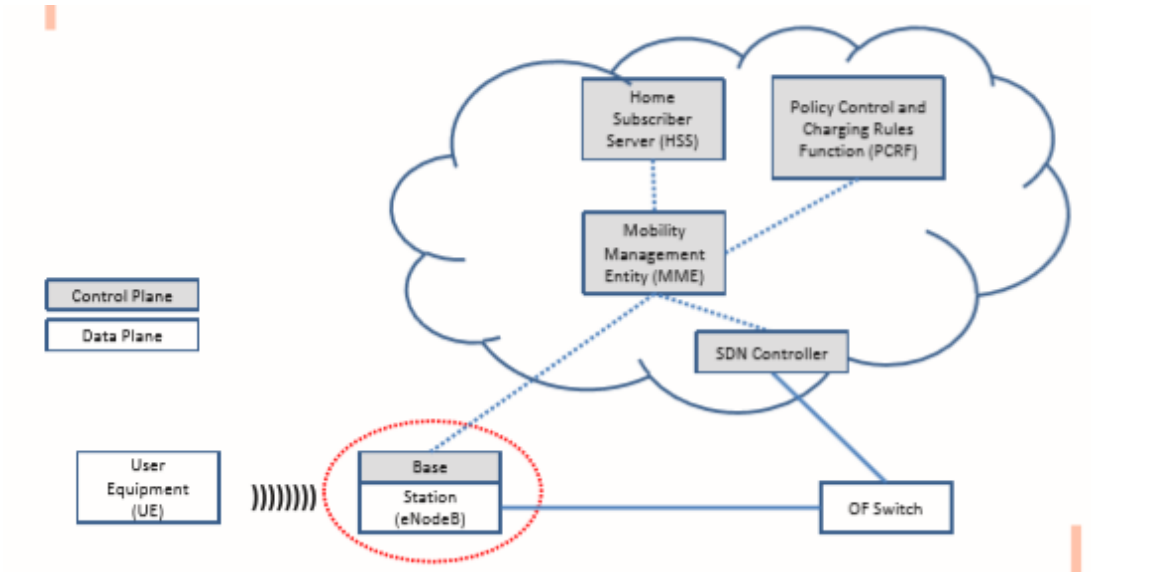


Εικόνα 3. 7: Δεύτερη επιλογή για την ενσωμάτωση του SDN στην LTE αρχιτεκτονική αποτελείτε από την ζεύξη του SDN controller με το MME

Αυτή η επιλογή επιτρέπει στον SDN controller να λαμβάνει γεγονότα κινητικότητας απευθείας από το MME που επιτρέπουν την εφαρμογή νέων κανόνων στα switches για την βελτίωση μονοπατιών δρομολόγησης.

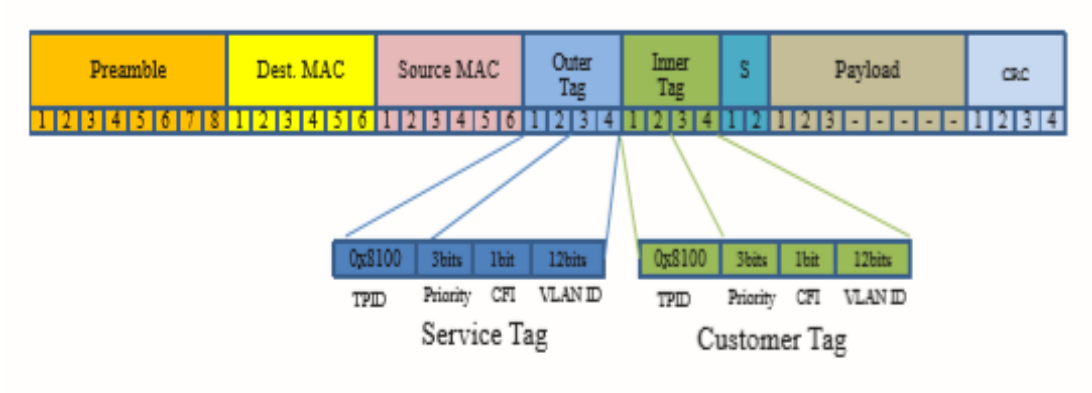
3.4.2 MIGRATION PATH OF SDN INTEGRATION IN LTE

Η ενσωμάτωση της λειτουργικότητας του SDN controller με το MME παρέχει έναν ομαλό τρόπο ενσωμάτωσης όπως επίσης μια διασπαστική λύση για τα κινητά δίκτυα. Το υπάρχον SDN πρέπει να κάνει ενεργές συγκεκριμένες απαιτήσεις όπου το επίπεδο δεδομένων βελτιστοποιείται για υψηλής ταχύτητας επεξεργασίας επιπέδου ροής με τη χρήση του OpenFlow. Στα SDMN το επίπεδο ελέγχου μετακινείται εκτός των βασικών στοιχείων του δικτύου σε κεντρικούς servers –αυτοί οι servers μοιάζουν με κλασικά σημεία εκκίνησης που υπάρχουν σε πολλά πρωτόκολλα κινητικότητας. Για τον λόγο αυτό, η πρόταση είναι να μετακινηθεί η λειτουργικότητα του controller και του S/PGW στα ίδια στοιχεία του δικτύου που βρίσκεται και η λειτουργικότητα του MME. Γι' αυτό, η λειτουργία των S/P-GW εξαφανίζεται και αντί αυτού ένα δίκτυο SDN βασισμένο σε πακέτα μεταγωγής χρησιμοποιείται. Αυτή η προσέγγιση προσθέτει ευελιξία και αξία στην δικτύωση και υποστηρίζει την σταδιακή εισαγωγή υψηλότερης διεκπαιρεωτικής ικανότητας του δικτύου, βέλτιστη διαχείριση ροών και δυνατότητες χειραγώγησης της κίνησης. Η επόμενη εικόνα δείχνει την ενσωμάτωση της κινητικότητας στον SDN controller και περιλαμβάνεται σαν μέρος του δικτυακού στοιχείου MME. Η κινητικότητα είναι βασική απαίτηση στα κινητά δίκτυα η οποία πρέπει να χειριστεί κατάλληλα με ελάχιστη καθυστέρηση και μειωμένη σηματοδοσία.



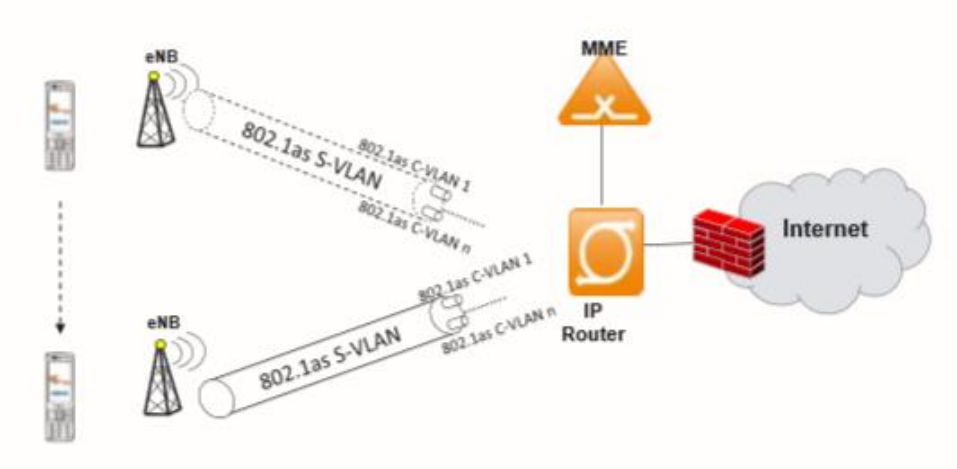
Εικόνα 3. 8: Φαίνεται ην ενσωμάτωση της κινητικότητας στον SDN controller και περιλαμβάνεται σαν μέρος του δικτυακού στοιχείου MME

Η κινητικότητα απαιτεί ειδική λειτουργία στα στοιχεία ενός κινητού δικτύου. Οι στενοί δεσμοί μεταξύ του MME και των SDN controllers επιτρέπουν στις λειτουργίες που σχετίζονται με χρονικούς περιορισμούς κινητικότητας χειρίζονται αποτελεσματικά από τον SDN controller. Αυτή η ενσωμάτωση παρέχει αποτελεσματική διαχείριση των handover στο SDMN.O OpenFlow controller προσθέτει και αφαιρεί τις flow entries από το flow table από την στιγμή που κάποιο handover γεγονός λαμβάνεται από το MME. Μια είσοδος στο flow table έχει τρία πεδία: ένα header πακέτου για να καθορίσει το flow, μια ενέργεια που καθορίζει την επεξεργασία του πακέτου, και τέλος τα στατιστικά. Εκτός από την ενσωμάτωση των MME και του SDN controller προτείνεται η χρήση του 802.1ad για να επιτρέψει διπλή ετικέτα στα Ethernet switches όπως φαίνεται παρακάτω.



Εικόνα 3. 9: *Διπλή ετικέτα στα Ethernet switches*

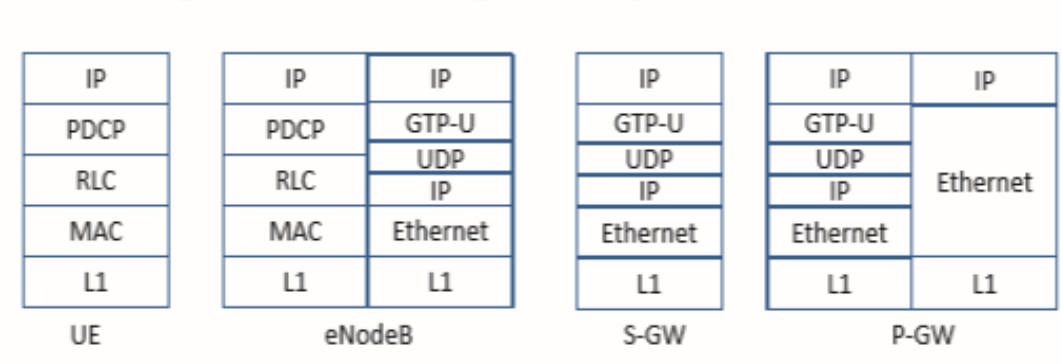
Οι διπλές ετικέτες μας επιτρέπουν να έχουμε μέχρι και 2^{12} ετικέτες υπηρεσιών σαν ένα εξωτερικό τούνελ και 2^{12} ετικέτες πελατών για εσωτερικά τούνελ (π.χ. σύνολο 2096 εσωτερικών και εξωτερικών τούνελ). Αυτά τα εξωτερικά VLANS μπορούν να χρησιμοποιηθούν για την εγκαθίδρυση τούνελ μεταξύ eNodeBs και IP router που βρίσκονται στο ίδιο Ethernet τμήμα για να παρέχουν πρόσβαση στο Internet (εικόνα 3.10). Τα εξωτερικά VLANS που εγκαθιδρύονται από τα eNodeBs μπορούν να τοποθετηθούν σε διαφορετικά Mobile Virtual Network Operators (MVNO). Τα 2^{12} εσωτερικά τούνελ μπορούν να εξυπηρετήσουν μέχρι και 10 MVNO σε μια περιοχή από 400 eNodeBs.



Εικόνα 3. 10: Τα εξωτερικά VLANS μπορούν να χρησιμοποιηθούν για την εγκαθίδρυση τούνελ μεταξύ eNodeBs και IP router που βρίσκονται στο ίδιο Ethernet τμήμα για να παρέχουν πρόσβαση στο Internet

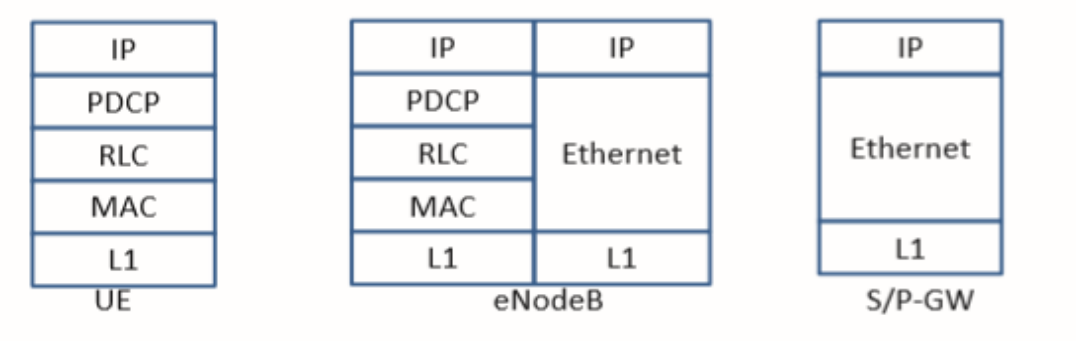
Επιπλέον, η ενσωμάτωση των λειτουργιών του ελεγκτή με το MME και S/P-GW σε ένα και μόνο στοιχείο δικτύου απλοποιεί την λειτουργία του δικτύου. Αυτό έχει ως αποτέλεσμα την επόμενη διάσπαση όπου το επίπεδο δεδομένων διαχειρίζεται από ένα μόνο MME/Controller στοιχείο. Η εξέλιξη προς αυτήν την αρχιτεκτονική μπορεί να γίνει σταδιακά, όπου το MME θα διατηρεί τις τωρινές διεπαφές και θα λαμβάνει την σηματοδότηση μέσω της S1-MME διεπαφής. Το MME θα διατηρεί τις τωρινές διεργασίες και θα εγκαθιδρύει τα GTP τούνελ μεταξύ των eNodeB και S/P-GW. Ταυτόχρονα το MME μπορεί να συμπεριλαμβάνει μια καινούργια SDN λειτουργία που εγκαθιδρύει επικοινωνία μεταξύ eNodeB και IP router απευθείας στο επίπεδο 2 χωρίς GTP tunneling. Σε αυτό το σενάριο το ίδιο MME όταν λαμβάνει μια σηματοδότηση από το SDN eNodeB μέσω της S1-MME διεπαφής θα εγκαθιδρύσει την σύνδεση με το τερματικό SDN switch πάνω από το L2 με την χρήση TUN διεπαφών.

Η δικτυακή στοίβα που χρησιμοποιείται για το επίπεδο του χρήστη παρουσιάζεται στην παρακάτω εικόνα (εικόνα 3.11). Τα radio επίπεδα τερμ0ατίζονται στα eNodeB από όπου το GTP χρησιμοποιείται μέχρι το S-GW και το PGW που παρέχει την γέφυρα για το Internet.



Εικόνα 3. 11: Δικτυακή στοίβα που χρησιμοποιείται για το επίπεδο του χρήστη

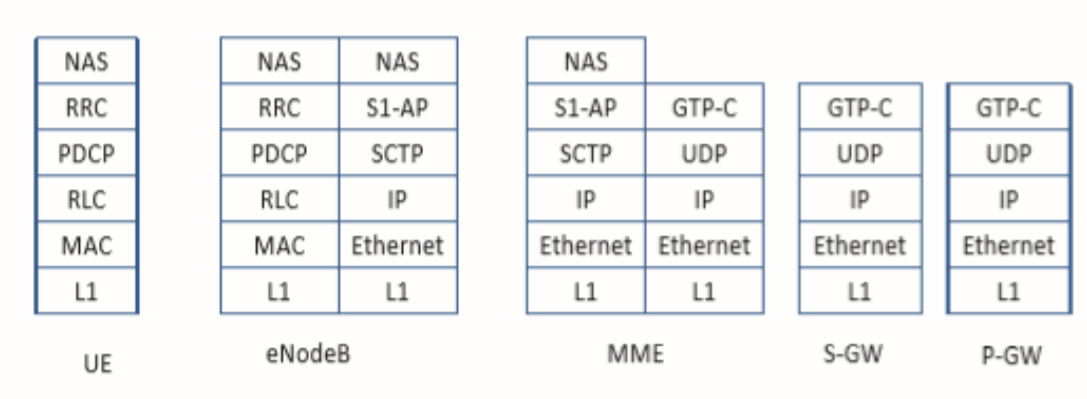
Η χρήση του 802.1ad στο backhaul και η ενσωμάτωση του MME με τον SDN OF controller επιτρέπει την αφαίρεση του GTP. Αυτό θα έχει ως αποτέλεσμα στην απλοποίηση της στοίβας στο eNodeB όπου τερματίζει τα radio επίπεδα και συμπεριλαμβάνει το Ethernet switch διαμέσου του υπόλοιπου δικτύου στο backhaul όπως φαίνεται στην επόμενη εικόνα. Επιπλέον, τα S/P-GW απλοποιούνται μετά την αφαίρεση του GTP και αποτελούνται από απλά Ethernet switch και IP router μέχρι το Internet. Σε αυτήν την αρχιτεκτονική η κινητικότητα πραγματοποιείται μέσω του SDN controller.



Εικόνα 3. 12: Απλοποίηση της στοίβας στο eNodeB όπου τερματίζει τα radio επίπεδα και συμπεριλαμβάνει το Ethernet switch διαμέσου του υπόλοιπου δικτύου στο backhaul

Αυτή η κινητικότητα οδηγεί σε ένα βέλτιστο δίκτυο μεταφοράς(transport network) όπως επίσης σε κλιμακωτό επίπεδο ελέγχου που συγκλίνει σε ένα μοναδικό στοιχείο του δικτύου το MME με ενσωματωμένες τις λειτουργίες του SDN OF controller. Αυτό το MME θα τρέχει είτε σε αποκλειστικό HW ή σε υπηρεσίες cloud για να επιτρέπει πολλαπλά instances που χρειάζεται για να ξεπεράσουμε την scalability του να έχουμε όλες τις λειτουργίες σε ένα μόνο στοιχείο του δικτύου.

Το MME από την άλλη μεριά θα διατηρεί τη σημερινή του δικτυακή στοίβα όπως παρουσιάζεται στην εικόνα παρακάτω. Οι αλλαγές στο επίπεδο δεδομένων ενώ διατηρείται η σηματοδοσία στο MME για να υποστηρίξουμε κληροδοτημένο eNodeB επιτρέπει μια ομαλή μετάβαση. Το MME θα είναι ικανό να διαχειρίζεται τα τωρινά στοιχεία του δικτύου π.χ. τα eNodeB και τα S/P-GW αλλά η ενσωμάτωση με το SDN επιτρέπει την διαχείριση νέων eNodeB και S/P-GW όπου το GTP έχει αφαιρεθεί.



Εικόνα 3. 13: Το MME διατηρεί τη σημερινή του δικτυακή στοίβα

Επιπλέον στην ενσωμάτωση των MME με το SDN και την απλοποίηση του backhaul δικτύου με την αφαίρεση του GTP από τα στοιχεία του δικτύου (π.χ. eNodeB και S/P-GW), το δίκτυο πρέπει να γίνει πιο επίπεδο. Τα στοιχεία του δικτύου τυπικά τοποθετούνται στο core network. Η ενσωμάτωση του MME με το SDN ακλουθώντας αυτήν την τοπολογία δεν είναι scalable.

Αντί αυτού τα δίκτυα πρέπει να είναι πιο επίπεδα ώστε τα στοιχεία του δικτύου να τοποθετούνται όσο πιο κοντά στα eNodeB στο backhaul. Αυτό επιτρέπει την ανάπτυξη access networks που μπορούν να σταθούν μόνα τους με τα δικά τους στοιχεία δικτύου. Ο συντονισμός των πολλαπλών access networks θα γίνεται με την χρήση κεντρικής Database και το handover μεταξύMME τοποθετημένο σε κάθε access network γίνεται μέσω της διεπαφής S10.

Η δικτυακή στοίβα σηματοδότησης παραμένει η ίδια για την επικοινωνία με κληροδοτούμενα στοιχεία του δικτύου LTE όπως το S/P-GW και άλλα MMEs. Επιπλέον στην απλούστευση της αρχιτεκτονικής του LTE, το ζήτημα κλιμάκωσης πρέπει να λυθεί. Η προτεινόμενη λύση είναι η μετακίνηση των δικτυακών στοιχείων του LTE στην νέα αρχιτεκτονική του MME μαζί με τον SDN OF controller στο access networks. Το MMEs σε διαφορετικά access networks μπορεί να μιλά με κάθε άλλο με την standard διεπαφή. Σε αυτήν την προσέγγιση τα MMEs σε διαφορετικά access network θα έχουν πρόσβαση στα HSS που βρίσκονται στο core network.

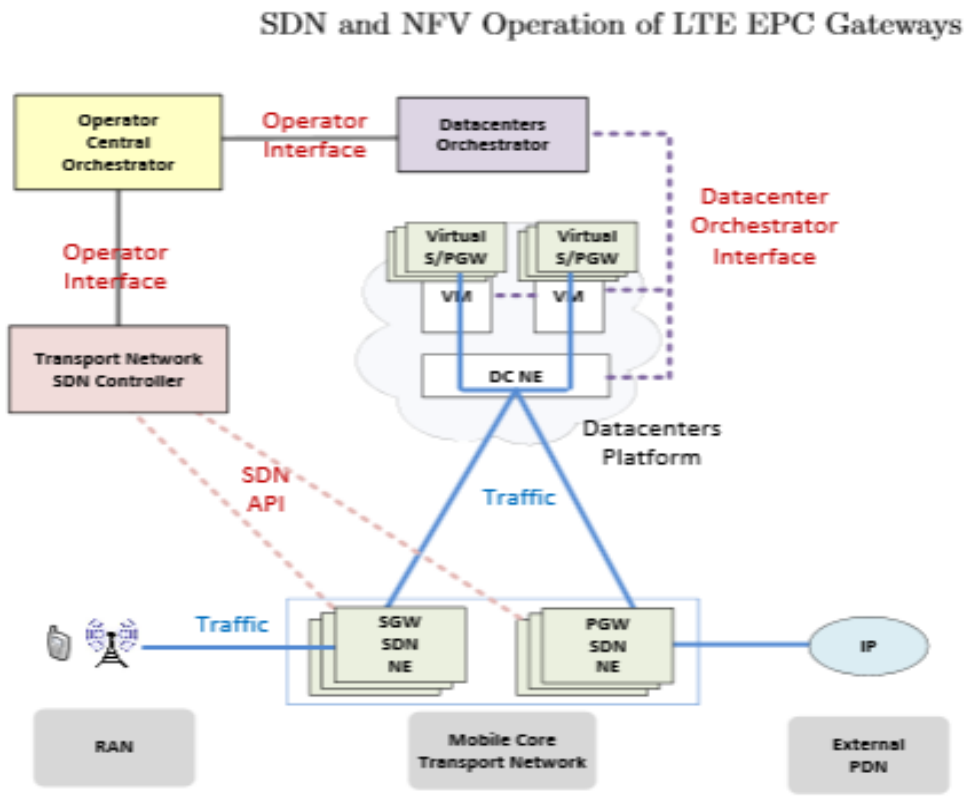
Αυτή η πιο επίπεδη αρχιτεκτονική επιτρέπει την χρήση του 802.1ad για κάθε access networks παρέχοντας την απαραίτητη απλοποίηση της LTE αρχιτεκτονικής ενώ παρέχει κομμάτια του δικτύου για virtual operators με αποκλειστικά VLANs

3.4.3 Η λειτουργία του SDN και του NFV στο LTE EPC

Το SDN μαζί με το NFV επιτρέπει, όπως περιγράψαμε παραπάνω, στους διαχειριστές των δικτύων να ελέγχουν τους πόρους του δικτύου με διακεκριμένο τρόπο και να εφαρμόζουν δυναμικές αλλαγές στο δίκτυο.

Η ευελιξία στη τοποθέτηση των πόρων παρέχεται από το NFV, όπου τα στοιχεία του δικτύου φιλοξενούνται σαν εικονικά στοιχεία και μπορούν να μετακινηθούν σε διαφορετικά εικονικά περιβάλλοντα. Η Ευελιξία στο επίπεδο του ελέγχου παρέχεται από το SDN, όπου η κίνηση του δικτύου μπορεί να σχηματίζεται δυναμικά από έναν κεντρικοποιημένο ελεγκτή.

Σύμφωνα με την εργασία[23] προτείνεται η παρακάτω αρχιτεκτονική για την χρήση των τεχνολογιών του SDN και NFV σε LTE EPC Getways:



Εικόνα 3. 14: Αρχιτεκτονική για την χρήση των τεχνολογιών του SDN και NFV σε LTE epc getways

Η αρχιτεκτονική εφαρμόζει το SDN και το virtualization στο Mobile core networks gateways για να επιτευχθεί δυναμική τοποθέτηση πόρων σε συνάρτηση με τις απαιτήσεις της κίνησης. Το μοντέλο που προτείνεται μεταμορφώνει το υπάρχον mobile core networks gateway σε virtual instances που φιλοξενούνται από datacenter πλατφόρμες και SDN στοιχεία δικτύου (π.χ. OpenFlow Switches) στο δίκτυο μεταφοράς

Το NFV μεταμορφώνει τις υπάρχουσες mobile gateways (SGW,PGW) σε εικονικές instances λογισμικού που τρέχουν πάνω σε εξοπλισμό του εμπορείου. Με αυτόν τον τρόπο ο διαχειριστής του δικτύου μπορεί να

λειτουργεί τα core gateways σε επίπεδο datacenter όπου παρέχεται η ευέλικτη και δυναμική τοποθέτηση των υπολογιστικών πόρων του datacenter δικτύου κατάλληλα σε κάθε virtualized instance.

Μέσα στο δίκτυο μεταφοράς, κάθε πύλη μπορεί να αντικατασταθεί από SDN NE(network elements), τα οποία είναι υπεύθυνα για την μεταφορά και την καθοδήγηση της κίνησης που προέρχεται από το access δίκτυο ή εξωτερικό δίκτυο δεδομένων και προορίζεται για τα virtual instances στο datacenter.

Στην αρχιτεκτονική που προτάθηκε χρειάζονται αρκετά στοιχεία ελέγχου και ενορχήστρωσης.

- **Datacenters Orchestrator (DC-O)**: Απαιτείται για την τοποθέτηση επαρκών πόρων στα εικονικά mobile core gateways από έναν κοινό τόπο πόρων που βρίσκεται στην πλατφόρμα του datacenter και περιλαμβάνει τόσο δικτυακούς όσο και υπολογιστικούς πόρους. Τοποθετεί την υπολογιστική ισχύ, όπως τους πυρήνες των επεξεργαστών, την μνήμη ή τον αποθηκευτικό χώρο στα gateways instances, οι οποίοι πόροι πρέπει να είναι επαρκής για να διαχειρίζονται τους μεγάλους όγκους δεδομένων κίνησης και να πετυχαίνουν παρόμοια ή και καλύτερη απόδοση από την σημερινή αρχιτεκτονική η οποία είναι βασισμένη στο υλικό. Επιπλέον, ο orchestrator είναι υπεύθυνος για την εγκαθίδρυση της συνδεσιμότητας μεταξύ των φυσικών στοιχείων του datacenter, καθώς και με το εξωτερικό δίκτυο. Τέλος, χειρίζεται τον συγχρονισμό και την μετεγκατάσταση των εικονικών gateway instances σε περίπτωση που κάποια αλλαγή στο δίκτυο πραγματοποιείται από τον διαχειριστή(operator).

- **Transport SDN Controller (SDN-C):** Χρησιμοποιείται για τον έλεγχο του δικτύου μεταφοράς(transport network), δηλαδή ελέγχει ώστε τα SDN NEs να χειρίζονται την προώθηση της κίνησης προς τα εικονικά gateways. Παρέχοντας έλεγχο στο SDN δίκτυο μεταφοράς παρέχεται η δυναμική διαχείριση της κίνησης, καθώς με το SDN-C δίνεται η ικανότητα να προσαρμόζουμε και να μεταβάλουμε το setup του δικτύου κατά τον χρόνο εκτέλεσης. Το SDN-C διαμορφώνει το δίκτυο το οποίο μεταφέρει το υψηλότερο όγκο κίνησης, μεταξύ του access δικτύου και του δικτύου εξωτερικών πακέτων δεδομένων(π.χ. Internet). Διαβεβαιώνει ότι υπάρχουν κανόνες μέσα σε κάθε SDN NE για να καθοδηγήσουν την κίνηση στο datacenter που φιλοξένει το επιθυμητό gateway.

- **Operator Central Controller (OCC):** περιέχει την κεντρική λογική για να δίνει διαστάσεις στο δίκτυο και να αποφασίζει για τις αλλαγές του δικτύου. Οι αλλαγές στο δίκτυο υποκινούνται από πολλούς λόγους, για παράδειγμα την πρόσθεση νέων στοιχείων στο δίκτυο, το φορτίο του δικτύου ή την μείωση των ενεργειακών κοστών με το να κλείσουν μέρη του δικτύου συμπεριλαμβανομένου και datacenter. Το OCC επίσης συμπεριφέρεται σαν διεπαφή μεταξύ του DC-O και του SDN-C,όπου η διεπαφή μεταδίδει τις απαιτήσεις του διαχειριστή του δικτύου.

3.5 Λειτουργικότητα προτεινόμενης αρχιτεκτονικής

Η αρχιτεκτονική που προτείνεται στο [19], και όπως παρουσιάζεται στην εικόνα 3.15, σκοπεύει στην ελάχιστη αλλαγή της 3GPP αρχιτεκτονικής με σκοπό να ενσωματώσει τα core στοιχεία στο OpenFlow δίκτυο. Οι κύριες core διεπαφές: S1-MME,S1-U,S6a και Gx λειτουργίες διατηρούνται, όπως επίσης και η πιστοποίηση του UE, η εξουσιοδότηση και η intra-3GPP διαχείριση κινητικότητας που πραγματοποιείται από το Mobility Management Entity (MME). Ο μηχανισμός επιλογής του υπάρχοντος Serving Gateway (SGW) / Packet Data Network Gateway (PGW) βασίζεται στην αλλαγή των Domain Name System (DNS) βαρών. Το MME θα ρωτήσει τον OpenFlow controller μέσω της NorthBound διεπαφής Representational State Transfer - REST, Application Programming Interface - API) ότι είναι ικανό να εγκαθιδρύσει κανόνες προώθησης στα OpenFlow switches.Η λειτουργία των κύριων στοιχείων παρουσιάζεται παρακάτω:

• Ο SDN controller: Είναι η κεντρική οντότητα που είναι υπεύθυνη για την διαχείριση του επιπέδου χρήστη και για τις αποφάσεις δρομολόγησης μεταξύ του eNodeB User Plane (eNB-U) και S/PGW User Plane (S/PGW-U).

- (S/PGW-U): Μια από τις ευθύνες του SDN controller είναι να διατηρεί τα ανανεωμένα στατιστικά των πορτών, να ισορροπεί την κίνηση και να πραγματοποιεί προγραμματισμό των ροών. Ο controller ρωτάει περιοδικά τα bytes που εκπέμπονται σε κάθε switch και βασιζόμενο στις πληροφορίες που τα switch του επιστρέφουν, διανέμει ίσα το φορτίο. Για έναν default bearer, ο μηχανισμός προγραμματισμού των ροών υπολογίζει το μονοπάτι από την πηγή στον προορισμό βασιζόμενο στους περιορισμούς του bandwidth, το προφίλ του κάθε UE απαιτεί και εγκαθιδρύει νέες εισόδους ροών στα eNB-Us και S/PGW-U switches. Ο controller μπορεί να εγκαθιδρύσει κανόνες στα OpenFlow (OF) switches τόσο με τρόπο reactive όσο και proactive: κατά την διαρκεί της εγκαθίδρυσης default bearer, οι λειτουργίες του controller στην βρίσκονται στην proactive κατάσταση (συμπληρώνει τις εισόδους ροών εκ των προτέρων) ενώ, για τα dedicated bearers παίρνει την απόφαση από την πληροφορία που υπάρχει στην επικεφαλίδα του πακέτου (reactive mode).

• Το MME: Είναι υπεύθυνο για την επιλογή του S/PGW-U και ρωτάει τον controller μέσω της NorthBound διεπαφής χρησιμοποιώντας την Open API. Ένα από τα πλεονεκτήματα αυτής της λύσης είναι η υψηλή ελαστικότητα, το οποίο επιτρέπει στους φορείς εκμετάλλευσης κινητής τηλεφωνίας να διατηρούνται αυτόνομα MME ή να τα ενσωματώνουν στο cloud. Για παράδειγμα, μπορούν οι MME, S/PGW Control Plane(S/PGW-C) και Policy and Charging Rules Function(PCRF) εφαρμογές να τρέχουν σαν Virtual Machine (VM)s (π.χ.OpenStackNeutron) πάνω από τον controller και με αυτόν τον τρόπο η τυποποιημένη διεπαφή S1-MME (μεταξύ Evolved Node B (eNB) και MME), S6a (μεταξύ MME και HomeSubscriberServer(HSS)) και Gx (μεταξύ PGW και PCRF) μπορούν να διατηρούνται.

• S/PGW-C: Οι λειτουργίες ελέγχου του SGW και PGW όπως η τοποθέτηση του Tunnel Endpoint Identifier (TEID) πραγματοποιούνται από τον controller. Μια βάση δεδομένων απαιτείται με σκοπό να αποθηκεύσει τις Internet Protocol (IP) διευθύνσεις του UE μετά την ερώτηση του MME και την διατήρηση των bearer Quality of Service (QoS) παραμέτρων (μεταδίδονται από τον PCRF στο SWG! (SWG!)-C και στο προφίλ του χρήστη). Σε αυτήν την αρχιτεκτονική το επίπεδο ελέγχου και το GPRS Tunneling Protocol

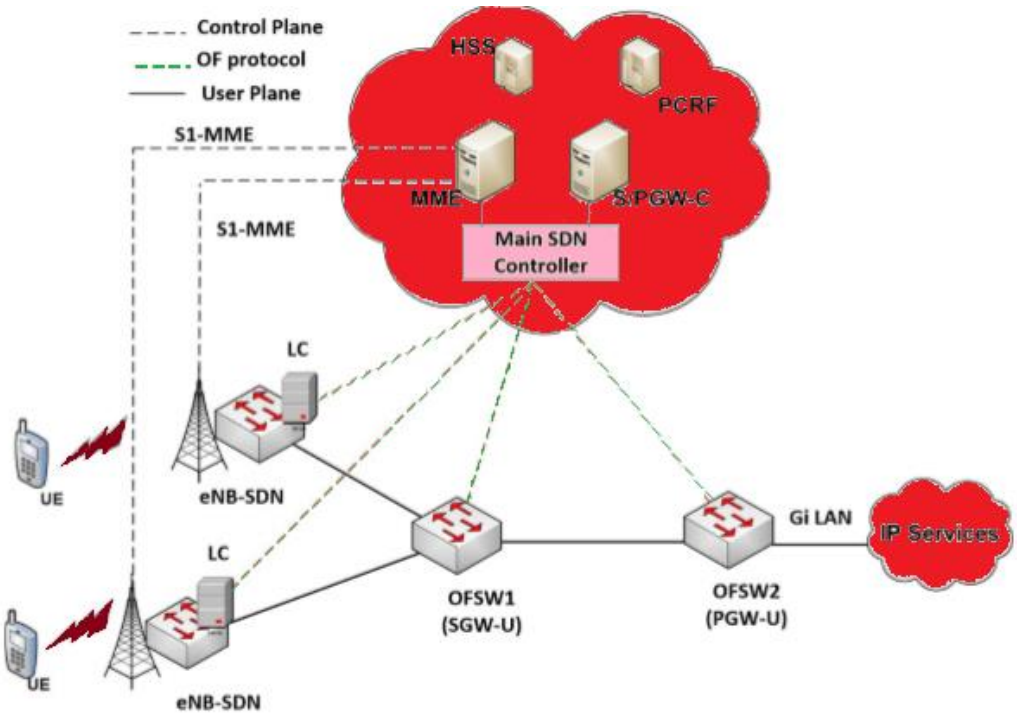
ControlPlane (GTP-C) διατηρείται με την έννοια ότι παρέχει διαφανή λειτουργικότητα στις διεπαφές ελέγχου του 3GPP standard.

• eNB-SDN/ S/PGW-U: Είναι OF switches τα οποία είναι ικανά να διαχειρίζονται εισόδους ροών που λαμβάνονται από τον controller στους πίνακες ροών. Αυτά τα switches είναι υπεύθυνα για την προώθηση κίνησης χρήστη μεταξύ του Evolved Node B (eNodeB) και των IP λειτουργιών (π.χ. Internet).

• Local Controller (LC): Είναι υπεύθυνο για την ταξινόμηση των πακέτων στα access switches με βάση την Location IP (LocIP) (η οποία αποτελείται από UEs IP διευθύνσεις και Base Station (BS) prefix). Επικοινωνεί με τον κεντρικό controller μέσω ενός Open API.

• Το eNodeB: Διατηρεί τις radio λειτουργίες που καθορίζονται στα 3GPP standards για την radio διεπαφή.

Στην εικόνα 3.15 παρουσιάζεται η αρχιτεκτονική που προτείνεται σε αυτήν την εργασία, η οποία δεν διαφέρει πολύ από την αρχιτεκτονική που προτείνεται στο [19].



Εικόνα 3. 15 : *Προτεινόμενη αρχιτεκτονική*

Στην εικόνα 3.15 μπορούμε να παρατηρήσουμε ότι δεν υπάρχει το GTP πρωτόκολλο μέσα στο EPC: στο επίπεδο χρήστη το GTP τούνελ αναγνωρίζεται μοναδικά από το ζεύγος των TEIDs (αντιστοιχούν στις διευθύνσεις των κόμβων πηγής και προορισμού) μαζί με τις IP διευθύνσεις πηγής και προορισμού και τον αριθμό της πόρτας του User Datagram Protocol (UDP). Στην παραδοσιακή αρχιτεκτονική του EPC,κάθε

bearer αναγνωρίζεται από το τούνελ και ένα UE μπορεί να έχει πολλαπλά sessions που αντιστοιχούν σε πολλαπλούς bearers. Από την στιγμή που εφαρμόζεται το OpenFlow στην αρχιτεκτονική, ένας OF switch είναι η πλατφόρμα που χρησιμοποιείται για την επεξεργασία όλης της κίνησης. Παρόλα αυτά, το OpenFlow πρωτόκολλο στην πιο πρόσφατη έκδοση του (OpenFlow 1.5.0) δεν υποστηρίζει GTP ταίριασμα, ούτε TEID στην επικεφαλίδα του GTP.

Υπάρχουν δύο πιθανές λύσεις για το πρόβλημα. Μία επιλογή είναι το OpenFlow πρωτόκολλο να υποστηρίζει GTP τούνελ και TEIDs .Μια άλλη πιθανότητα είναι να διατηρήσουμε το OpenFlow ανέγγιχτο και να προσθέσουμε μια επιπλέον οντότητα (πχ μια line card) υπεύθυνη για GTP tunneling. Πρέπει να λειτουργεί ως 'decapsulator' και μπορεί να τοποθετηθεί μεταξύ του eNode Band και του eNB-U switch. Η κύρια λειτουργία αυτής της οντότητας είναι να αφαιρεί τα TEIDs. Με αυτόν το τρόπο, το δίκτυο μεταξύ των eNBs και του Internet είναι IP core που μπορεί εύκολα να διαχειριστεί από τον controller.

Από την εικόνα 3.11 μπορούμε να παρατηρήσουμε στην GTP στοίβα ότι δεν υπάρχει επικεφαλίδα Ethernet μετά την UDP επικεφαλίδα. Μετά το decapsulation του GTP, το πακέτο δεν θα έχει πληροφορία επιπέδου 2, για να εφαρμόσουμε το GTP πρωτόκολλο στο Open vSwitch. Για τον λόγο αυτό σε περίπτωση που θέλουμε να επεκτείνουμε το Open vSwitch να μπορεί να υποστηρίζει GTP πρωτόκολλο, η πληροφορία του επιπέδου 2 πρέπει να προστεθεί.

Μια πιθανή λύση σε αυτό το πρόβλημα είναι να δημιουργήσουμε μια απλή επικεφαλίδα Ethernet και τα πακέτα θα προωθούνται σύμφωνα με πληροφορία επιπέδου 3, από τη στιγμή που η πληροφορία επιπέδου 2 δεν υπάρχει.

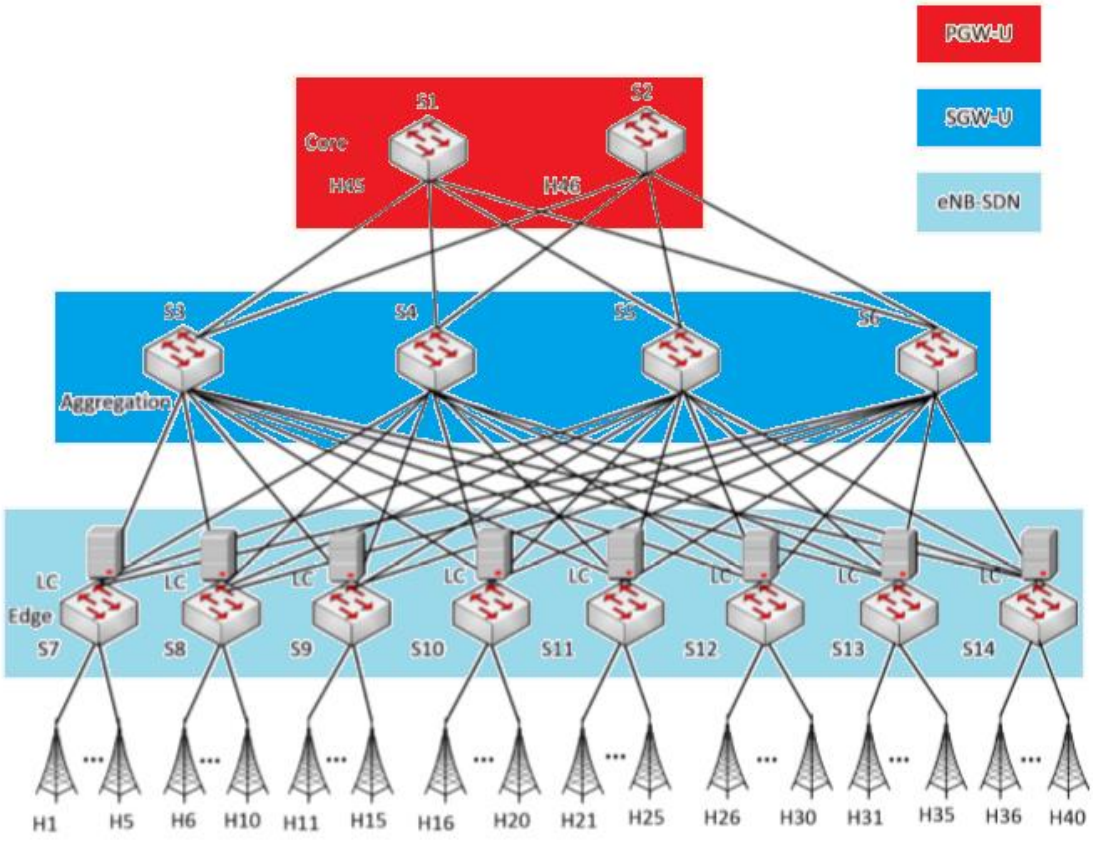
Μια άλλη επιλογή είναι απλά να δρομολογήσουμε τα πακέτα βασιζόμενοι στις πληροφορίες που ήδη υπάρχουν. Με αυτόν τον τρόπο, εκμεταλλευόμαστε πλήρως τις ικανότητες του Open vSwitch καθορίζοντας μόνο τις πόρτες εισόδου και εξόδου. Η απόδοση αυτής της λύσης αναμένεται να είναι καλύτερη σε θέματα ρυθμαπόδοσης από την λύση να ενεργοποιήσουμε επιπέδου 3 αντιστοιχία και να τροποποιήσουμε την επικεφαλίδα του Ethernet. Το mapping επιπέδου 3 και η αλλαγή της επικεφαλίδας Ethernet θα αυξήσει το overhead του GPRS Tunneling Protocol User Plane (GTP-U) τούνελ. Αυτές οι λύσεις έχουν μελετηθεί στο[28] και βασιζόμενοι στην απόδοση σε αυτήν την εργασία θα επιλέξουμε την πρώτη λύση, δηλαδή την επιλογή να δρομολογήσουμε τα πακέτα βασιζόμενοι στις πληροφορίες που ήδη υπάρχουν, για την εφαρμογή του GTP-U στο Open vSwitch.

3.6 Επιλεγμένη τοπολογία

Από τη στιγμή που δεν υπάρχουν σωστές ή λάθος τοπολογίες, οι πλειοψηφία των διαχειριστών των δικτύων θα επιλέγουν τα δίκτυα τους βασιζόμενοι σε γεωγραφικές και επιχειρηματικές απαιτήσεις. Παρόμοιο με το μοντέλο των κέντρων δεδομένων, μια κινητή τοπολογία αποτελείται από τρία επίπεδα: το πρόσβασης(access),της συσσωμάτωσης (aggregation) και του πυρήνα (core). Στο [29] παρουσιάζονται διαφορετικές τοπολογίες και ο συνδυασμός τους. Μια mobile backhaul τοπολογία όπως ένας δακτύλιος παρέχει καλύτερη αντοχή και χωρητικότητα απ' ότι μια τοπολογία δέντρου, επίσης είναι περισσότερο

scalable και πετυχαίνει χαμηλότερα επίπεδα latency. Για τον λόγο αυτό η τοπολογία δακτυλίου επιλέγεται για την σύνδεση των eNBs, ενώ για το mobile core δίκτυο σετάρεται full-mesh(πλήρες πλέγμα) configuration για μεγαλύτερη αντοχή και σκοπούς πλεονασμού(redundancy).

Στην εικόνα 3.16 που ακολουθεί παρουσιάζεται η τοπολογία που ακολουθήσαμε



Εικόνα 3. 16: *Τοπολογία fat tree*

Στην προτεινόμενη τοπολογία έχουμε 2 core switches, 4 aggregation switches και 8 edge switches συνδεμένα το καθένα σε 5 hosts. Τα core switches έχουν το ίδιο αριθμό από ζεύξεις όπως ο αριθμός των aggregation switches, όπου τα aggregation switches έχουν τον ίδιο αριθμό από ζεύξεις όπως τα edge switches (εικόνα 3.16).Εάν x είναι ο αριθμός των core switches, x^2 είναι ο αριθμός των switches που βρίσκονται στο επίπεδο aggregation, και x^3 είναι ο αριθμός των edge switches.Αυτήν την τοπολογία την συναντάμε συχνά σε data centers, λόγω της αντοχής που παρουσιάζει.

Σκοπός της τοπολογίας είναι να εξομοιώσει το mobile core δίκτυο που αποτελείται από 40 eNBs συνδεδεμένα σε 8 edges witches. Ο αριθμός των edge switches είναι ίσος με τον αριθμό των LocalController(LC) switch που σχηματίζουν την Location IP (LocIP), μοναδική για κάθε χρήστη. Στην προτεινόμενη τοπολογία κάθε eNB έχει μόνο μια LocIP, όπως UE IP και Base Station (BS) prefix. Τα 4

aggregation switches αντικαθιστούν τις λειτουργίες του SGW user plane με εικονικά OpenFlow switches. Σε αντίθεση με την κοινή ανάπτυξη του EPC, όπου ο αριθμός των SGWs είναι ίσος με τον αριθμό των PGWs, στην προτεινόμενη τοπολογία, ο αριθμός τν aggregation switches αντιστοιχεί στους κόμβους SGWείναι διπλάσιος από τα core switches.

Για την ανάπτυξη της τοπολογίας χρησιμοποιήθηκε η γλώσσα Python μέσα στο directory /mininet/custom του mininet. Το σκριπτακι Fattreetopo.py αποτελείται από κώδικα που μπορεί κάποιος να το βρει σε διάφορες κοινότητες στο internet, αλλά έχει τροποποιηθεί για να ταιριάξει στις ανάγκες αυτής της εργασίας.

3.7 Πολυπλοκότητα Εφαρμογής

• Επίπεδο Ελέγχου: Η πολυπλοκότητα της λύσης που προτάθηκε μπορεί να αξιολογηθεί σε επίπεδο εργαλείων που απαιτούνται για την εφαρμογή της, το scalability και το κόστος. Η προτεινόμενη λύση διατηρεί την λειτουργικότητα του επιπέδου ελέγχου, από τη μεριά του implementation, απαιτεί τόσο hardware όσο και software αλλαγές. Για παράδειγμα, οι MME, S/PGW-C και οι PCRF λειτουργίες μπορούν να τρέχουν σαν VMs στο OpenStack Neutron (σε κέντρα δεδομένων) και μπορούν να επικοινωνούν με τον κεντρικό ελεγκτή μέσω του REST API. Από την στιγμή που ο ελεγκτής και το OpenStack μπορούν να εφαρμοστούν σε πλατφόρμα virtualized server, δραστικά μειώνεται η πολυπλοκότητα του υλικού που βρίσκεται στο δίκτυο ενός operator. Για παράδειγμα, ένας controller μπορεί να συνδεθεί με εκατοντάδες switches και να διαχειριστεί εκατομμύρια αιτήσεις το δευτερόλεπτο.

• Επίπεδο Δεδομένων: Το καινοτόμο user plane αποτελείται από πολλαπλά OpenFlow switches με ενσωματωμένο Open vSwitch (OVS). Το Open vSwitch [31] είναι ένα εικονικό switch το οποίο υποστηρίζει ροές (flows), Virtual LAN (VLAN)s, trunking, QoS, port aggregation και Layer 3. Με την λύση αυτή απλοποιείται το user plane GTP encapsulation στα παραδοσιακά EPC με την εφαρμογή της Layer 2 προώθησης και του flow aggregation βασισμένο στο ToS. Η διεύθυνση LocIP προσθέτει τα flows που έρχονται από το UE και το eNB σε ένα και μόνο tag και διαφοροποιεί μεταξύ user sessions βασισμένο στις πόρτες προορισμού. Επίσης εγγυάται το scalability του δικτύου. Ένα από τα πλεονεκτήματα της εφαρμογής που παρουσιάστηκε παραπάνω είναι το χαμηλό κόστος εφαρμογής της λύσης, από την στιγμή που οι τιμές για το υλικό του OpenSwitch είναι μερικά χιλιάδες Euro σε σχέση με τα εκατομμύρια που πληρώνουν οι operators για την εφαρμογή και το configuration του κάθε gateway.

Κεφάλαιο 4°

4.1 Mininet

Το Mininet είναι ένας εξομοιωτής δικτύου. Τρέχει μια συλλογή από end-hosts, switches, routers, και ζεύξεις σε έναν απλό Linux πυρήνα. Χρησιμοποιεί απλό virtualization για να κάνει ένα απλό σύστημα να μοιάζει με ολόκληρο δίκτυο που τρέχει στον ίδιο πυρήνα, σύστημα, και κώδικα χρήστη. Ένας Mininet host συμπεριφέρεται όπως ακριβώς μια πραγματική μηχανή, μπορείς να κάνεις ssh σε αυτόν και να τρέξεις αυθαίρετα προγράμματα (συμπεριλαμβανομένου οτιδήποτε είναι εγκατεστημένο κάτω από το σύστημα των Linux.) Τα προγράμματα που τρέχεις μπορούν να στέλνουν πακέτα μέσω διεπαφή που μοιάζει με την πραγματική Ethernet διεπαφή, με ταχύτητα ζεύξης και καθυστέρηση. Τα πακέτα επεξεργάζονται από αυτό που μοιάζει με αληθινό Ethernet switch, router, ή middlebox, με συγκεκριμένο ποσό πακέτων στην ουρά. Όταν δύο προγράμματα , όπως ένας iperf client και server, επικοινωνούν μέσω του Mininet, η μετρήσιμη απόδοση πρέπει να ταιριάζει αυτή του native μηχανήματος.

Σε συντομία, οι εικονικοί host του Mininet, switches, links, και controllers είναι οι πραγματικοί –απλά δημιουργούνται με την χρήση λογισμικού και όχι υλικού – και το μεγαλύτερο μέρος της συμπεριφοράς τους είναι παρόμοιο με αυτό των συσκευών υλικού. Είναι συχνά πιθανό να δημιουργήσουμε δίκτυα Mininet τα οποία μοιάζουν με δίκτυα υλικού, ή ένα δίκτυο υλικού το οποίο μοιάζει με Mininet δίκτυο, και να τρέξουν τον ίδιο κώδικα και εφαρμογές.

Υπάρχουν πολλοί λόγοι για να επιλέξουμε το mininet σαν εργαλείο εξομοίωσης. Πρώτον, υπάρχουν λίγες συσκευές δικτύου με σκοπό την εφαρμογή του SDN standard καθώς δεν είναι μια ευρέως διαδεδομένη τεχνολογία από βιομηχανικής άποψης. Επιπλέον η εφαρμογή ενός δικτύου με μεγάλο αριθμό συσκευών είναι κοστοβόρα υπόθεση. Το mininet μπορεί να εξομοιώνει διαφορετικά στοιχεία του δικτύου, όπως host, switches,routers και ζεύξεις.

Μερικά από τα Θετικά του Mininet

1. Είναι γρήγορο – το να ξεκινήσεις ένα απλό δίκτυ παίρνει λίγα δευτερόλεπτα.
2. Μπορούμε να δημιουργήσουμε όποιες τοπολογίες επιθυμούμε: ένα απλό switch, μεγαλύτερες τοπολογίες, κέντρο δεδομένων ή οτιδήποτε επιθυμούμε.
3. Μπορούμε να τρέξουμε πραγματικά προγράμματα: οτιδήποτε τρέχει σε Linux είναι διαθέσιμο, από web servers σε TCP window monitoring εργαλεία μέχρι Wire shark.
4. Μπορούμε να τροποποιήσουμε τον τρόπο προώθησης των πακέτων: Τα switches του Mininet είναι προγραμματιζόμενα με την χρήση του OpenFlow πρωτοκόλλου.
5. Μπορούμε να τρέξουμε το Mininet στο laptop, σε έναν server,ή σε VM.
6. Μπορούμε να μοιραστούμε και να αναπαράγουμε αποτελέσματα: καθένας με υπολογιστή μπορεί να τρέξει το πρόγραμμα μας.
7. Μπορεί να χρησιμοποιηθεί εύκολα: μπορούμε να δημιουργήσουμε και να τρέξουμε ε Mininet πειράματα γράφοντας απλά ή σύνθετα Python scripts.
8. Το Mininet είναι open source.

4.1.1 Περιορισμοί του Mininet

Το Mininet έχει κάποιους περιορισμούς, όπως:

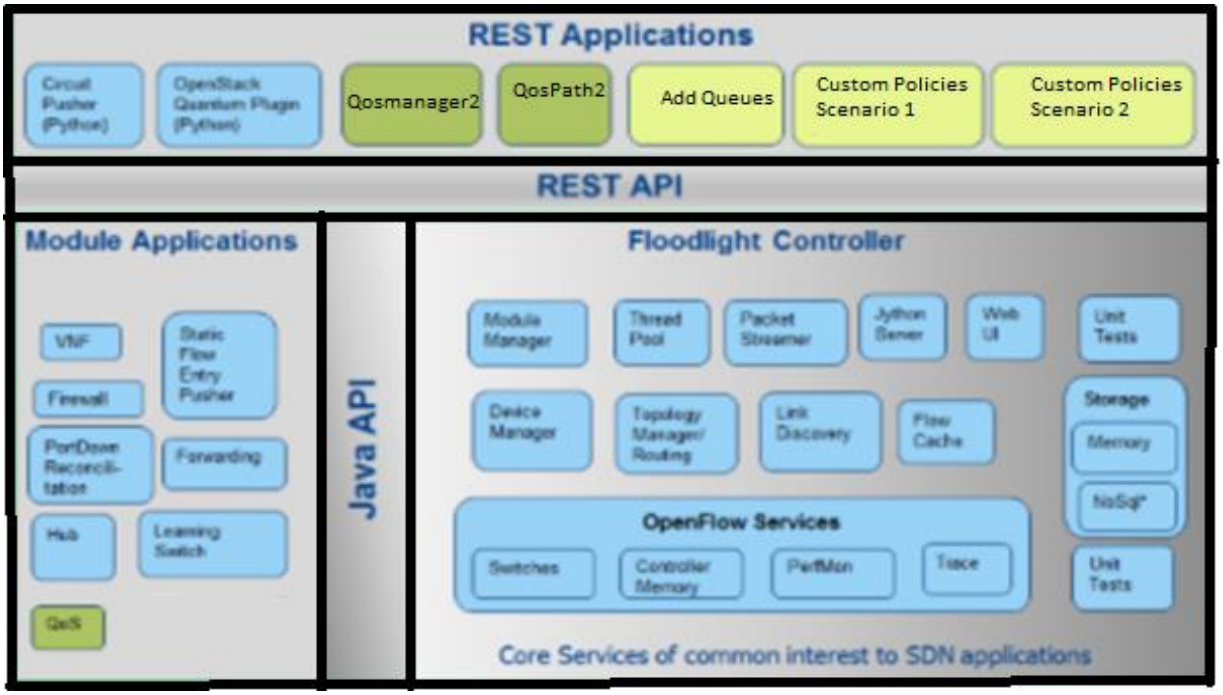
- Υπάρχουν περιορισμένοι πόροι καθώς τρέχουμε σε ένα σύστημα: Εάν ο server έχει 3 GHz CPU και μπορεί να κάνει μεταγωγή 10 Gbps προσομοιωμένης κίνησης, αυτοί οι πόροι θα πρέπει να μοιραστούν και να ισορροπήσουν μεταξύ των virtual hosts και switches.
- Το Mininet χρησιμοποιεί έναν απλό Linux πυρήνα για όλους τους εικονικούς hosts;αυτό σημαίνει ότι μπορούμε να τρέχουμε λογισμικό βασισμένο σε BSD, Windows,ή άλλο λειτουργικό σύστημα.
- Το Mininet δεν θα γράψει μόνο του τον OpenFlow controller, εάν χρειαζόμαστε τροποποιημένη δρομολόγηση ή συμπεριφορά μεταγωγής, θα χρειαστεί να βρούμε ή να αναπτύξουμε χαρακτηριστικά που επιθυμούμε στον controller.
- Το Mininet δίκτυο είναι απομονωμένο από το LAN και από το internet. Παρόλα αυτά μπορούμε να χρησιμοποιήσουμε το NAT για να συνδέσουμε το δίκτυο Mininet στο LAN μέσω Network Address Translation. Μπορούμε επίσης να προσδέσουμε πραγματική ή εικονική διεπαφή υλικού στο Mininet δίκτυο.

4.1.2 Χαρακτηριστικά του mininet

- Ευελιξία, νέες τοπολογίες και νέα χαρακτηριστικά μπορούν να τοποθετούνται στο λογισμικό, χρησιμοποιώντας γλώσσες προγραμματισμού και κοινά λειτουργικά συστήματα.
- Διαδραστικότητα. Η διαχείριση και η εκτέλεση του εξομοιωμένου δικτύου πρέπει να συμβαίνει σε πραγματικό χρόνο, όπως συμβαίνει στα πραγματικά δίκτυα.
- Ρεαλισμός, καθώς η συμπεριφορά του πρωτότυπου πρέπει να αντιπροσωπεύει την πραγματική συμπεριφορά με μεγάλο βαθμό εμπιστοσύνης, έτσι ώστε οι εφαρμογές και τα πρωτόκολλα να χρησιμοποιούνται χωρίς μεταβολές του κώδικα.
- Τα πρωτότυπα δίκτυα που δημιουργούνται πρέπει εύκολα να μοιράζονται με άλλους, οι οποίοι μπορούν να τρέξουν και να τροποποιήσουν τα πειράματα.
- Scalability, τα πρότυπα δίκτυα που δημιουργούνται πρέπει να μπορούν να εφαρμόζονται σε μεγάλη κλίμακα με εκατοντάδες switches και στοιχεία δικτύου.

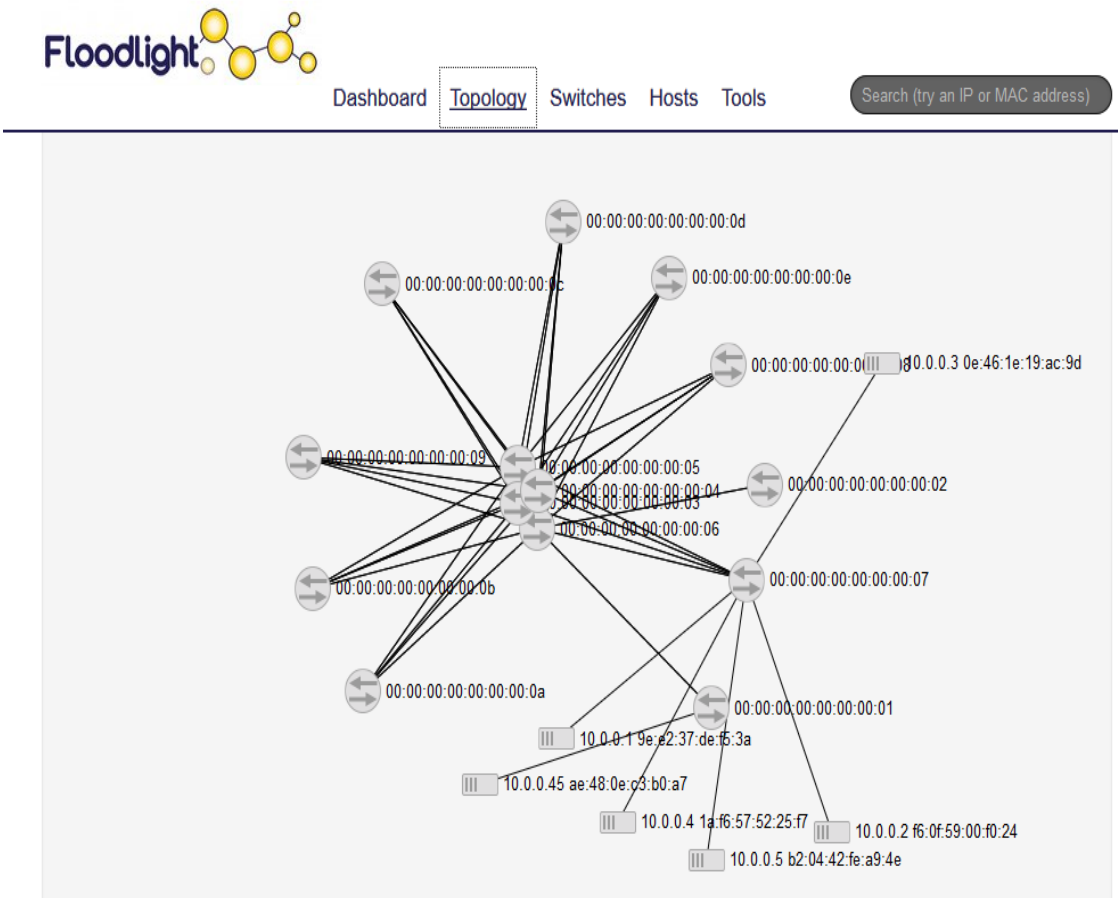
4.2 Floodlight Controller

Ο Floodlight SDN Controller είναι ένα SDN open-source project που υποστηρίζεται από τη Big Switch Networks και από την κοινότητα του SDN. Άλλες εταιρείες και οργανισμοί που έχουν συνεισφέρει στην ανάπτυξη των APIs του Floodlight controller είναι οι: Arista, Brocade, Citrix, Dell, Extreme Networks, Fujitsu, Google, HP, IBM, Intel, Juniper Networks και η Microsoft. Το Floodlight είναι ένας OpenFlow controller με Apache άδεια, πιο ώριμο από άλλους controller όπως το OpenDaylight, που είναι επίσης αναπτυγμένο σε γλώσσα Java. Υποστηρίζει ένα μεγαλύτερο εύρος από virtual switches όπως το Open vSwitch, όπως και φυσικά OpenFlow switches που παρέχουν σύνδεση μεταξύ των OpenFlow και μη-OpenFlow δικτύων [34]. Στην εικόνα-4.1 βλέπουμε τις εσωτερικές σχέσεις στον Floodlight controller, τις μονάδες(modules) εφαρμογών που είναι γραμμένες σε Java και τις εφαρμογές που είναι γραμμένες πάνω από τον Floodlight. Αυτές οι εφαρμογές επικοινωνούν με τον controller μέσω REST API. Αυτό που δείχνει η εικόνα είναι ότι οι μονάδες με μπλε χρώμα αντιπροσωπεύουν τα modules του Floodlight που υπάρχουν στην έκδοση που χρησιμοποιήσαμε για τις μετρήσεις. Τα modules με πράσινο χρώμα συμπεριλαμβάνει τις QoS εφαρμογές και τα κίτρινα κουτιά αντιπροσωπεύουν τα επιπλέον modules που αναπτύχθηκαν στα πλαίσια αυτής της εργασίας.



Εικόνα 4. 1: Floodlight modules

Η τοπολογία μας εφαρμόζεται στο Mininet και είναι συνδεδεμένο με τον Floodlight controller στο 192.168.56.103. Ο Floodlight επιτρέπει και την οπτικοποίηση της τοπολογίας με ένα GUI Web Interface (Εικόνα 4.2). Εκτός από το τμήμα Topology, το τμήμα Tools δείχνει την έξοδο από την εφαρμογή των μηχανισμών διασφάλισης του QoS (Εικόνα 4.3).



Εικόνα 4. 2: Τοπολογία

		ip-dest=167772205, ip-src=167772171, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,		Enqueue 0:1	null	32000
1738022603	SDN11.00:00:00:00:00:00:00:09	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772171, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:09	Enqueue 0:4	null	32000
733109952	SDN6.00:00:00:00:00:00:00:01	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772166, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:01	Enqueue 0:2	null	32000
822470109	SDN6.00:00:00:00:00:00:00:06	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772166, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:06	Enqueue 0:1	null	32000
1687600358	SDN6.00:00:00:00:00:00:00:08	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772166, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:08	Enqueue 0:4	null	32000
2070332257	SDN1.00:00:00:00:00:00:00:01	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772161, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:01	Enqueue 0:2	null	32000
1744289877	SDN1.00:00:00:00:00:00:00:06	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772161, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:06	Enqueue 0:1	null	32000

Εικόνα 4. 3:Οι πολιτικές Qos εμφανίζονται στον τομέα Tools

Η μονάδα QoS (εικόνα 4.3) επιτρέπει μέσω της REST API στις πολιτικές διασφάλησης του QoS να εφαρμοστούν στο δίκτυο. Αυτή η μονάδα επιτρέπει στο configuration του ορίου του ελάχιστου και μέγιστου ρυθμού καθώς και στις Diff Serv Type of Service (ToS) μονάδες, καθορίζει κανόνες QoS και εφαρμόζει πολιτικές σε ένα μονοπάτι.

Η μονάδα QoS αποτελείται από τρία κύρια αρχεία Python: το "CircuitPusher.py" ,το"QosManager2.py" και το "QoSPath2". Αυτά τα σκριπτάκια έχουν τροποποιηθεί από την αρχική τους μορφή ώστε να καλύψουν τις ανάγκες αυτής της εργασίας [35] και εκτός από αυτά έχουν εφαρμοστεί και επιπλέον σκριπτάκια, όπως τα mininet_add_queues.py ,custom_policies_sc1.py, custom_policies_sc2.py. Οι προδιαγραφές του OpenFlow 1.0 και OpenFlow 1.3 υποστηρίζουν την ώθηση των QoS καταστάσεων στα switches. Το QoS module είναι ανεπτυγμένο σε Java και έρχεται με ένα σύνολο από Python scripts που επιτρέπουν διαφορετικά configurations. Οι πολιτικές που προστίθενται στο μονοπάτι βασίζονται στα ToS byte που αντιστοιχούν σε κάθε session.

4.3 Εξομοίωση κίνησης σε SDN-based τοπολογία στο επίπεδο χρήστη

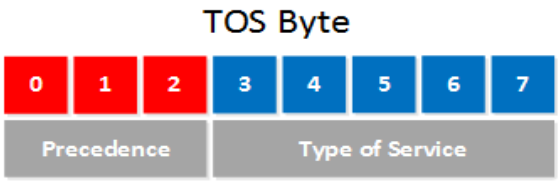
Το Mininet είναι το εργαλείο που θα χρησιμοποιήσουμε για να εξομοιώσουμε το δίκτυο της προτεινόμενης τοπολογίας και για να τρέξουμε δύο σενάρια, τα οποία βασίζονται στην εφαρμογή πολιτικών στους hosts, οι οποίοι θα στείλουν κίνηση, ταυτόχρονα, στους εξυπηρετητές. Σκοπός του τεστ είναι η εξομοίωση ενός κινητού δικτύου που αποτελείται από 40 Evolved Node B (eNB)s με πολλαπλούς χρήστες συνδεδεμένους. Οι συσκευές χρήστη (UE) θα αποστείλουν διαφορετικά sessions την ίδια στιγμή, τα session είναι βίντεο Voice over IP (VoIP) και Http. Τα δεδομένα είναι στο επίπεδο του χρήστη(user plane).

Ο controller που θα χρησιμοποιήσουμε για να πραγματοποιήσουμε τα πειράματά μας είναι ο Floodlight controller. Η επιλογή βασίστηκε στο γεγονός ότι παρέχει εργαλεία και QoS modules για τη διαχείριση της κατάστασης των QoS σε ένα OpenFlow δίκτυο. Τα modules που χρησιμοποιήσαμε για την πραγματοποίηση τω πειραμάτων βασίζονται στον κώδικα που παρέχεται από τον Ryan Wallner στο github, κάτω από free licence για να εφαρμόσουμε τις απαραίτητες τροποποιήσεις.

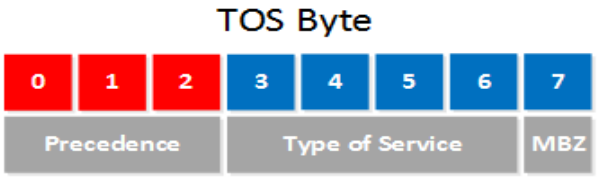
4.3.1 DSCP και ToS

Τα πακέτα που εισέρχονται στο επίπεδο του Differentiated Services(DiffServ) μπορούν να ταξινομηθούν με βάση πολλές παραμέτρους : IP διευθύνσεις πηγής και προορισμού, πρωτόκολλα επιπέδου 4 (TCP/UDP) και αριθμούς πορτών, τις τιμές των byte του πεδίου Type of Service (ToS) ή από πληροφορία επιπέδου 2. Από την στιγμή που τα πακέτα ταξινομηθούν, σύμφωνα με τιμές από τουλάχιστον μια από αυτές τις παραμέτρους, μπορούν να επεξεργαστούν. Ο ρόλος του κεντρικού controller είναι να κατηγοριοποιήσει την κίνηση σε διαφορετικές τάξεις και να εφαρμόσει παραμέτρους QoS σε αυτές τις τάξεις.

Η αρχιτεκτονική του DiffServ καθορίζει το πεδίο του DiffServ, το οποίο αντικαθιστά το πεδίο ToS στο IPv4 για να πραγματοποιήσει τις αποφάσεις για Per-Hop Behavior (PHB) σχετικά με την ταξινόμηση πακέτων και σχετικά με λειτουργίες των όρων της κίνησης, όπως η μέτρηση, η βαθμονόμηση και οι πολιτικές που εφαρμόζονται. Για να το πετύχουν αυτό, τα πακέτα πρώτα χωρίζονται σε κλάσεις με το σετάρισμα των κατάλληλων bit στο ToS byte στην IP επικεφαλίδα. Το αρχικό IPv4 ToS καθορίζεται στο IETF RFC 1349 και αποτελείται από 4 bit πεδία. Στο RFC 2474,το ToS αποτελείται από το Differentiated Services Code Point (DSCP) (τα πρώτα 6 bits) στα οποία προσθέτονται 2 μηδενικά bits στο τέλος για να σχηματίσουν ένα ολόκληρο ToS byte.



Εικόνα 4. 4: Tos byte καθορισμένο στο IETF RFC 1349



Εικόνα 4. 5: Tos byte καθορισμένο στο IETF RFC 2474

Οι βασικές κλάσεις που καθορίζονται από το DiffServ είναι η ”default” ή Best Effort (BE), η Expedited Forwarding (EF) και η AssuredForwarding(AF). Η Ef χρησιμοποιείται για ”strict” priority (π.χ.video και voice), και η AF χρησιμοποιείται για business differentiation. Τα πακέτα μπορούν να μαρκάρονται με τυχαία DSCP τιμή, που αντιστοιχεί στην κατάλληλη AF, EF ή BE κλάση. Στο 3GPP, το QoS Class Indicator (QCI) αντιστοιχείται απευθείας στο DSCP που καθορίζεται στο RFC 2474.

1. Best Effort (Προκαθορισμένη PHB, καθορίζεται στο RFC 2474): Το default PHB καθορίζει ότι ένα πακέτο έχει τιμή DSCP το ‘000000’ και παίρνει την υπηρεσία BE από κόμβο που είναι συμβατός με το DS (ένας κόμβος που συμμορφώνεται με τις απαιτήσεις του DiffServ).Μια υπηρεσία BE μπορεί να σχετίζεται με πρωτόκολλο Hypertext Transfer Protocol(HTTP)

Επειδή η BE κίνηση με ίδια πόρτα προορισμού ”80” μπορεί να έρχεται από διαφορετικά eNBs με διαφορετικά tags, πρέπει να βελτιστοποιήσουμε την χρήση των πόρων των gateways. Για τον λόγο αυτό, ένα BE σύστημα πρέπει να δίνει σε όλους τους χρήστες ευκαιρίες για να πάρουν μερίδιο από το διαθέσιμο bandwidth.

2. Expedited Forwarding PHB (EF) : Το EF είναι το κύριο κλειδί στο DiffServ, γιατί παρέχει χαμηλή απώλεια πακέτων, χαμηλή καθυστέρηση, χαμηλό jitter, και βέβαιη υπηρεσία bandwidth. Εφαρμογές όπως η Voice over IP (VoIP), video, και προγράμματα online trading απαιτούν ευσταθή συμπεριφορά δικτύου. Το Expedited forwarding μπορεί να εφαρμοστεί χρησιμοποιώντας queuing με προτεραιότητα, μαζί με περιορισμό του κόστους σε μια κλάση. Παρότι το EF, όταν εφαρμόζεται σε δίκτυα DiffServ, παρέχει υψηλής ποιότητας υπηρεσία, πρέπει να έχει στόχο μόνο κρίσιμες εφαρμογές ,επειδή όταν εμφανίζεται συμφόρηση, δεν μπορεί να διαχειρίζεται όλη, ή την περισσότερη κίνηση, με υψηλή προτεραιότητα. Το Expedited Forwarding είναι χρήσιμο για εφαρμογές όπως το VoIP που απαιτεί πολύ χαμηλή απώλεια πακέτων, εξασφαλισμένο bandwidth, χαμηλή καθυστέρηση και χαμηλό jitter. Οι προτεινόμενη τιμή

DSCP για το EF είναι η ’101110’=46. Τα ToS byte χρησιμοποιούν και τα οχτώ bits, ενώ το DSCP χρησιμοποιεί μόνο τα πρώτα έξι ψηφία. Το EF μοτίβο θα γίνει ‘10111000’=184 όταν μιλάμε για ολόκληρο το οκτέτο.

3.Το Assured Forwarding PHB AF, καθορίζεται στο IETF, RFC2597: Το AF διαχωρίζει την κίνηση σε τέσσερις τάξεις, όπου κάθε AF τάξη εγγυάται έναν ελάχιστο αριθμό από πόρους (π.χ. χωρητικότητα και buffering). Μέσα σε κάθε κλάση, τα πακέτα τμηματοποιούνται περαιτέρω σε μια από τις τρείς κατηγορίες απώλειας. Σε περίπτωση συμφόρησης σε ένα κόμβο DiffServ σε μια συγκεκριμένη ζεύξη, τα πακέτα της συγκεκριμένης AFx κλάσης(π.χ.AF1) πρέπει να απορριφτούν, τα πακέτα στο AFxy θα χαθούν έτσι ώστε dP(AFx1) <= dP(AFx2) <= dp(AFx3), όπου dP(AFxy) είναι η πιθανότητα όπου τα πακέτα στην AFxy κλάση θα χαθούν. Η AFx κλάση μπορεί να συμβολίζεται από DSCP ‘xyzab0,’ όπου ‘xyz’ είναι 001/010/011/100 και ‘ab’ αντιπροσωπεύει το χάσιμο των bits προτεραιότητας. Παράδειγμα, τα πακέτα στο AF13 θα χαθούν πριν από τα πακέτα στο AF12 και πριν από τα πακέτα στο AF11.Το επίπεδο του forwarding assurance ενός IP πακέτου εξαρτάται από τους πόρους που δεσμεύονται για την AF κλάση, το τωρινό φορτίο της AF κλάσης, και την προτεραιότητα απώλειας του πακέτου[2].

4.3.2 Πρώτο Σενάριο

Στο πρώτο πείραμα οι hosts (h1-h5) που αντιστοιχούν στα eNB συνδέονται στο switch 7, στην τοπολογία που παρουσιάσαμε παραπάνω, και στέλνουν κίνηση στον εξυπηρετητή h45 μέσω του S1(PGW-U). Για την επιλογή του μονοπατιού που θα ακολουθηθεί χρησιμοποιείται ο αλγόριθμος Dijkstra. Το module του floodlight controller στο οποίο υπάρχει ο αλγόριθμος είναι το TopologyInstance.java. Στην εργασία μας γίνεται διάσχιση όλων των switches του επιπέδου συνέννοσης. Στο πείραμά μας μόνο ένα από τα switches του aggregation επιπέδου επιλέγεται και η κίνηση περνάει μόνο μέσω αυτού. Οι πολιτικές έχουν εφαρμοστεί μεταξύ του client και του server. Στα παρακάτω κομμάτια κώδικα παρουσιάζουμε αυτές τις πολιτικές.

Η μεταβλητή host1 αποθηκεύει τιμές από το 1 έως το 40 με εύρος 5,άρα παίρνουμε τον πρώτο host. Η μεταβλητή host2 μας δίνει τον δεύτερο host και η μεταβλητή host3 μας δίνει τους hosts τρία ,τέσσερα και πέντε.

hosts1 = range(1,40,5)

hosts2 = range(2,40,5)

rest = list(set(range(1,41))−set(hosts1)−set(hosts2))

ip_hosts1=[]

Στην συνέχεια τοποθετούμε την ip στον κάθε host

[2] T. Szigeti, C. Hattingh, R. Barton, and K. Briley, “End-To-End QoS Network Design: Quality of Service for Rich-Media and Cloud Networks.” Pearson Education, 2013

```
 for h in range(len(hosts1)):
    ip_hosts1.append("10.0.0."+str(hosts1[h]))
print ip_hosts1


ip_hosts2=[]


for h in range(len(hosts2)):
    ip_hosts2.append("10.0.0."+str(hosts2[h]))
print ip_hosts2


ip_rest=[]


for h in range(len(rest)):
    ip_rest.append("10.0.0."+str(rest[h]))
print ip_rest
```

Η πολιτική που εφαρμόζεται στον κάθε host φαίνεται από το παρακάτω κομμάτι κώδικα κατά το οποίο δίνουμε στον πρώτο host Differentiated Services Code Point (DSCP) ίσο με 34, σε class AF41, το οποίο αντιστοιχεί σε Type of Service ίσο με 136 και έχει περιορισμό στο bandwidth στα 30 Mbps. Θέλουμε να χάνουμε, δηλαδή, όσο το δυνατόν λιγότερα πακέτα, εφόσον η κίνηση δεν ξεπεράσει το όριο των 30 Mbps.

```
dscp = "34"
queue = "0"
for h in range(len(ip_hosts1)):
    cmd = "sudo ./qospath2.py --add --name SDN"+str(hosts1[h])+" -S "+ip_hosts1[h]+" -D "+ip_server+
" -J '{\"tos\":\""+dscp+"\",\"eth-type\":\"0x0800\",\"protocol\":\"17\",\"queue\":\""+queue+"\",\"priority\":\"32000\"}'"
```

Ο δεύτερος host θέλουμε να μεταφέρει κίνηση που αντιστοιχεί σε VoIP, άρα θα έχουμε ισχυρές απαιτήσεις υπηρεσιών για τον λόγο αυτό θα χρησιμοποιήσουμε DSCP με τιμή ίση με 46 που αντιστοιχεί στο Expedited Forwarding (EF) Per-Hob Behavior (PHB). Στο RFC 3246 σημειώνεται το εξής για το EF PHB «The intent of the EF PHB is to provide a building block for low loss, low delay, and low jitter services».

```
dscp = "46"

queue = "1"

for h in range(len(ip_hosts2)):

   cmd = "sudo ./qospath2.py --add --name SDN"+str(hosts2[h])+" -S "+ip_hosts2[h]+" -D"+ip_server+" -J '{\"tos\":\""+dscp+"\",\"eth-type\":\"0x0800\",\"protocol\":\"17\",\"queue\":\""+queue+"\",\"priority\":\"32767\"}'"
```

Οι hosts τρία, τέσσερα και πέντε έχουν dscp ίσο με 0, άρα πακέτα μπορούν να χαθούν και έχουν χαμηλότερο priority στην χρήση του bandwidth σε σχέση με τα πακέτα που στέλνουν οι προηγούμενοι hosts.

```
dscp = "0"

queue = "2"

for h in range(len(ip_rest)):

   cmd = "sudo ./qospath2.py --add --name SDN"+str(rest[h])+" -S " +ip_rest[h]+" -D"+ip_server+" -J '{\"tos\":\""+dscp+"\",\"eth-type\":\"0x0800\",\"protocol\":\"6\",\"queue\":\""+queue+"\",\"priority\":\"30000\"}'"
```

Το ether Type : 0x0800 σημαίνει ότι χρησιμοποιούμε πρωτόκολλο IPv4

Κάθε queue θα έχει διαφορετική πολιτική, για τον λόγο αυτό τα πακέτα που στέλνει ο host1 ανήκουν στην queue 0 ο host2 στην queue 1 οι host3, host4 και host5 στην queue 2. Σε κάθε πόρτα κάθε switch σετάρονται οι παραπάνω QoS queues, στην queue 0 έχουμε 30 Mbps, queue 1 έχουμε 9 Mbps και queue 2 έχουμε 6 Mbps. Οι πολιτικές δρομολόγησης σύμφωνα με το QoS επιστρέφουν το μικρότερο μονοπάτι μεταξύ της πηγής και του προορισμού.

Το OpenvSwitch χρησιμοποιεί την ιδιότητα του ”traffic control” του Linux για περιορισμό του ρυθμού. Στο κώδικα παρακάτω παρουσιάζεται το configuration που χρησιμοποιούμε, μέγιστο ρυθμό 1 Gbps (109 bits per second) και έχουν καθοριστεί τα QoS queues όπως περιγράφηκαν παραπάνω. Το πρώτο queue, q0

έχει το μέγιστο bandwidth (*min-rate=30000000 other-config:max-rate =30000000*), όπου το Q1 έχει bandwidth ρυθμό 9Mbps και Q2 έχει 6Mbps:

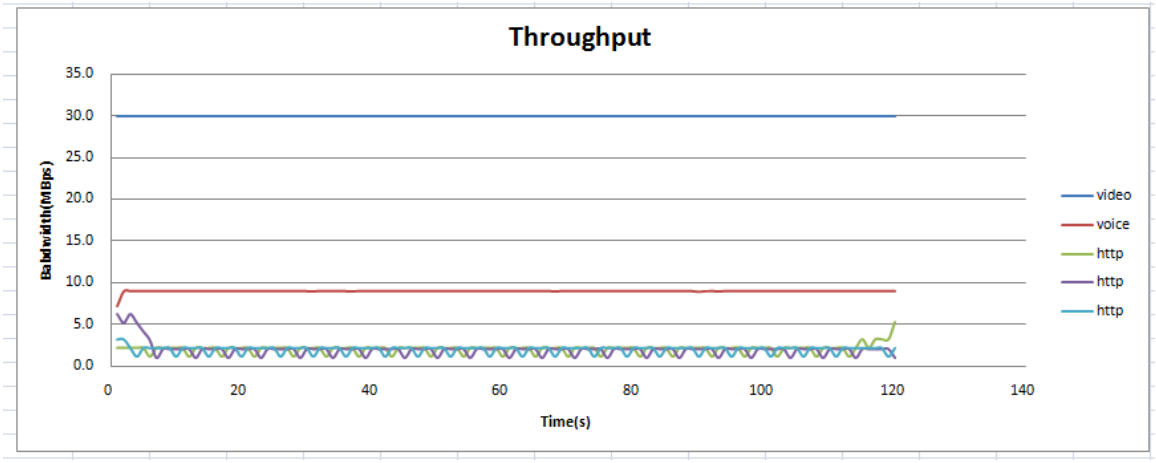
queuecmd = "sudo ovs-vsctl \%s -- --id=@defaultqos create qos type=linuxhtb other-config:max-rate=1000000000 queues=0=@q0,1=@q1,2=@q2 -- --id= @q0 create queue other-config:min-rate=30000000 other-config:max-rate =30000000 -- --id=@q1 create queue other-config:max-rate=9000000 other -config:min-rate=9000000 -- --id=@q2 create queue other-config:maxrate= 6000000 other-config:min-rate=6000000"

Το Protocol έχει τιμή 6 που αναφέρεται στο TCP και το 17 αναφέρεται στο πρωτόκολλο UDP.

Το priority ανά πακέτο είναι αντίστοιχα 32000,32767,30000

4.3.2.1 Ρυθμαπόδοση

Όπως έχουμε προαναφέρει, όλοι οι clients στέλνουν κίνηση ταυτόχρονα στον server με περιορισμό του bandwidth σύμφωνα με τις πολιτικές που έχουν εφαρμοστεί. Στην παρακάτω εικόνα φαίνεται ότι η video session (H1) λαμβάνει 29,5 Mbps, η voice session (H2) receives 8.8 Mbps και κάθε HTTP session λαμβάνει περίπου 2 Mbps (και τα τρία session μαζί στα 6 Mbps). Για την σύνδεση των τριών HTTP, η ρυθμαπόδοση (throughput)μειώνεται λόγω των ταυτόχρονων αιτήσεων που στέλνονται στον ίδιο server, και μοιράζονται το bandwidth των 6 Mbps.



Εικόνα 4. 6: Throughput

Το τεστ πραγματοποιήθηκε με την χρήση του εργαλείου Iperf τόσο για το User Datagram Protocol (UDP) όσο και για τη Transmission Control Protocol (TCP) κίνηση.

Έχουμε τοποθετήσει τις αντίστοιχες ToS τιμές, σε διαρκεί σύνδεσης 120 δευτερολέπτων και μέγεθος παραθύρου 256 k(UDP) και 128 k(TCP). Για το UDP το μέγεθος παραθύρου περιορίζει το μέγιστο μέγεθος του δεδομενογράμματος που μπορεί να ληφθεί με το να καθορίζει το μέγεθος του buffer.

Οι τιμές της ρυθμαπόδοσης που λάβαμε στο πρώτο τεστ επιβεβαιώνουν τις τιμές περιορισμού του ρυθμού για κάθε session (π.χ. 30 Mbps στην περίπτωση του video, 9 Mbps για φωνή και 6 Mbps bandwidth να μοιράζεται μεταξύ 3 HTTP συνδέσεων). Η TCP μοιράζεται όλο το διαθέσιμο bandwidth μεταξύ τους με τον μηχανισμό ελέγχου ροής (flow control mechanism). Για τον λόγο αυτό, ακόμα και αν η UDP ροή μειώσει τον ρυθμό μετάδοσης ή το τμηματοποιήσεις, το διαθέσιμο bandwidth θα καταναλωθεί από την TCP σύνδεση. Επιπλέον, όταν οι TCP συνδέσεις μοιράζονται έναν κόμβο που έχει γίνει bottleneck, η εξέλιξη των παραθύρων συμφόρησης μπορεί να συγχρονιστεί. Ως αποτέλεσμα, η ροή UDP επηρεάζεται σημαντικά από τον TCP συγχρονισμό. Αυτός είναι ο λόγος που η απόδοση στο χάσιμο των UDP πακέτων μπορεί να βελτιωθεί, ακόμα και αν ο ρυθμός μετάδοσης των UDP μειώνεται μετά την τμηματοποίηση. Η απώλεια των UDP πακέτων καθορίζει την εμπειρία του τελικού χρήση σε audio και video εφαρμογές, όπως η VoIP και τα video sessions. Τα υψηλά ποσοστά απώλειας μπορούν επίσης να προκαλέσουν υψηλότερο jitter και καθυστερήσεις στην UDP κίνηση. Οι VoIP δεν μπορούν να ξανασταλθούν, γιατί η πληροφορία δεν θα είναι σύγχρονη. Σε αντίθεση με το RTP, το UDP δεν χρησιμοποιεί sequence numbers ή timestamps για επανασυγχρονισμό του δέκτη, ούτε multicasting. Το πρωτόκολλο UDP δεν εγγυάται μετάδοση δεδομένων. Η κίνηση UDP δεν γίνεται acknowledged. Αυτό σημαίνει ότι ο client μπορεί να στέλνει κίνηση με κάθε bandwidth χωρίς να ενδιαφέρεται πόσα πακέτα θα χαθούν. Η διαφορά μεταξύ UDP και TCP είναι ότι το TCP θα ξαναστείλει τα χαμένα πακέτα και το UDP όχι.

4.3.3 Δεύτερο Σενάριο

Στο δεύτερο τεστ κάθε host –που αντιστοιχούν στα eNB- στέλνει κίνηση που περιέχει περισσότερες της μίας πολιτικής στους servers(PGW-U). Ο host1(h1) έχει τρία sessions: ένα video, ένα φωνής και ένα HTTP, ο h2 έχει ένα session φωνής και ένα session HTTP, οι h3,h5,h5 έχουν από δύο HTTP sessions. Ο αλγόριθμος Dijkstra χρησιμοποιείται και σε αυτό το σενάριο από τον controller για την εύρεση του καλύτερου μονοπατιού. Η κίνηση πηγαίνει διαδοχικά μια στον server H45 και μια στον server H46. Αποτέλεσμα είναι ένας server να μην εξυπηρετεί μόνο ένα είδος κίνησης. Όποτε η επιλογή του switch στο aggregation level εξαρτάται από τον επιλεγμένο server. Άρα, ο floodlight controller επιλέγει το καλύτερο μονοπάτι για έναν server και όλη η κίνηση που αφορά αυτό το server κατευθύνεται από αυτό το μονοπάτι, σε περίπτωση προβλήματος του switch στο aggregation level επιλέγεται η καινούργια μικρότερη διαδρομή, και δρομολογείται η κίνηση από εκεί.

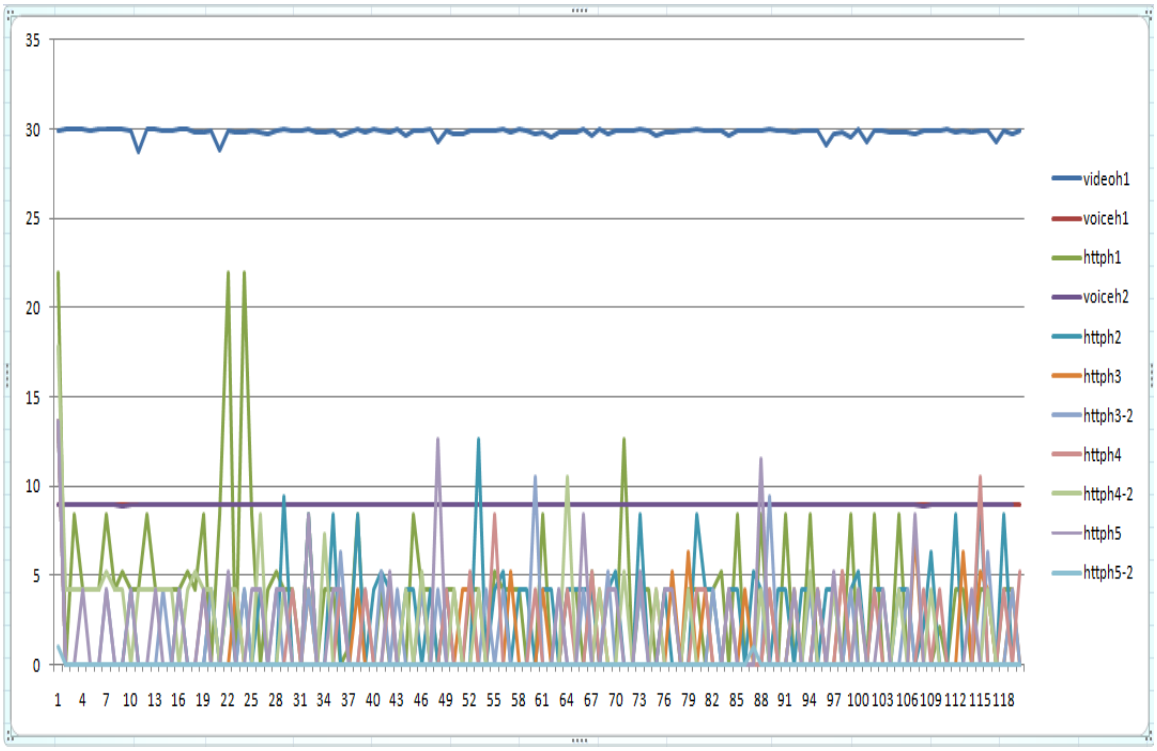
Τα χαρακτηριστικά του κάθε flow φαίνονται στον παρακάτω πίνακα

Client	Server	Protocol	Session	DSCP	ToS	Service	Queue	Bandwidth	Priority
H1	H45	UDP	1 Video	34	136	AF	0	30 Mbps	32000
H1	H46	UDP	1 Voice	46	136	EF	1	9 Mbps	32767
H1	H45	TCP	1 HTTP	0	0	BE	2	3.5 Mbps	30000
H2	H46	UDP	1 Voice	46	184	EF	1	8.8 Mbps	32767
H2	H45	UDP	1 HTTP	0	0	BE	2	2.5 Mbps	30000
H3	H46	TCP	1 HTTP	0	0	BE	2	1.15 Mbps	30000
H3	H45	TCP	1 HTTP	0	0	BE	2	1.5 Mbps	30000
H4	H46	TCP	1 HTTP	0	0	BE	2	1.5 Mbps	30000
H4	H45	TCP	1 HTTP	0	0	BE	2	2 Mbps	30000
H5	H46	TCP	1 HTTP	0	0	BE	2	1.5 Mbps	30000
H5	H45	TCP	1 HTTP	0	0	BE	2	0.5 Mbps	30000

Εικόνα 4. 7: Πίνακας 1

Όπως φαίνεται από τον πίνακα γίνεται προσπάθεια διαμοιρασμού της κίνησης μεταξύ του server h45 και h46.

Στην εικόνα που ακολουθεί μπορούμε να δούμε ότι υπάρχουν 8 HTTP συνδέσεις και κάθε μια παίρνει περίπου 2 Mbps από κάθε server, εκτός από την τελευταία σύνδεση που παίρνει 0.5 Mbps. Το bandwidth του video είναι περίπου 30 Mbps, ενώ τα δύο voice sessions έχουν περίπου 9 Mbps. Όλα τα TCP sessions καθορίζονται με μέγεθος παραθύρου ίσα με 1024 K. Σε αντίθεση με το πρώτο σενάριο, σε αυτό το τεστ τα UDP sessions δεν έχουν σεταρισμένο buffer size, και έτσι χρησιμοποιείται το προκαθορισμένο μέγεθος.



Εικόνα 4. 8:Ρυθμαπόδοση Σενάριο 2

Το χαρακτηριστικότερο αποτέλεσμα που μπορούμε να παρατηρήσουμε στην παραπάνω εικόνα είναι οι ακμές που υπάρχουν κατά την μεταφορά κίνησης HTTP, αυτές οι μεταβολές αντιστοιχούν στις διακυμάνσεις του εύρους ζώνης που πραγματοποιείται την ίδια χρονική στιγμή. Γενικά, αυτό που μπορούμε να παρατηρήσουμε για τις συνδέσεις TCP από την προηγούμενη εικόνα είναι ότι χάνονται πακέτα επειδή φτάνουν σε μια διεπαφή υπερπλήρη (congested). Όταν χάνονται τα πακέτα, οι αποστολείς μπορούν να καταλάβουν ότι το δίκτυο έχει συμφόρηση και τότε ο μηχανισμός του TCP, Slow start, ξεκινάει. Ο ρυθμός μετάδοσης μειώνεται απότομα μέχρι να μειωθεί η απώλεια πακέτων και στην συνέχεια ο ρυθμός μετάδοσης αυξάνεται σταδιακά. Όταν το TCP Slow Start ξεκινάει, όλοι οι αποστολείς του δικτύου κάνουν ένα βήμα πίσω και βλέπουμε μείωση του εύρους ζώνης. Σε περίπτωση που δεν έχουμε απώλεια πακέτων τότε οι αποστολείς στέλνουν πακέτα σε υψηλότερο ρυθμό, για τον λόγο αυτό στην παραπάνω εικόνα βλέπουμε και πολύ υψηλές κορυφές σε μερικές περιπτώσεις αποστολής πακέτων HTTP . Ο μέσος όρος του εύρους ζώνης παρουσιάζεται στον προηγούμενο πίνακα(2.1).

Σε αυτό το σενάριο ο h1 στέλνει ένα video session στον server h45 και ένα voice session στον server h46. Για παράδειγμα, το πρώτο voice session που προέρχεται από το h1 στέλνεται στον server h45, ενώ το δεύτερο voice session που μεταδίδεται από το h2 στέλνεται στο h46. Ως αποτέλεσμα, αν και είναι πιο σταθερό, το throughput μοιράζεται μεταξύ του video και των voice sessions. Οι απώλειες πακέτων που παρατηρούμε οφείλονται στον έλεγχο ροής του μηχανισμού του TCP. Από την άλλη μεριά το UDP είναι πιο απλό πρωτόκολλο χωρίς καθυστερήσεις κατά την σύνδεση, έλεγχο ροής, και αναμεταδόσεις. Το

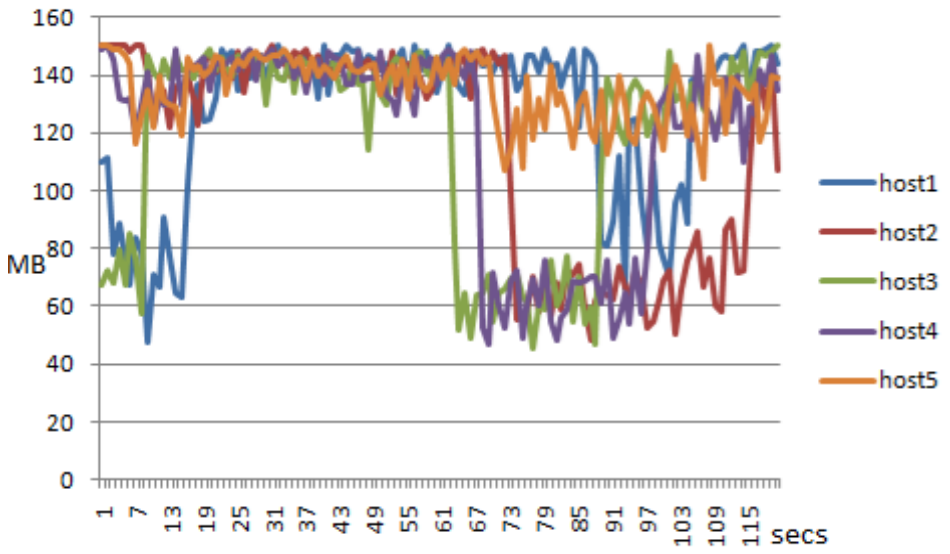
μέγεθος παραθύρου των 1024 kbps προκαλεί TCP congestion synchronization. Μειώνοντας το μέγεθος του παραθύρου στα 128 kbps παρατηρούμε πάλι κορυφές και απώλεια πακέτων.

4.3.4 Τρίτο Σενάριο

Σε αυτό το τεστ έχουμε σκοπό να προσομοιώσουμε την ταυτόχρονη αίτηση video από πέντε UE προς έναν εξυπηρετητή (h45). Τα video session που απαιτεί κάθε συσκευή χρήστη (h1-h5) μεταφέρονται με ρυθμό μετάδοσης 150Mbps. Ο όγκος των δεδομένων που μεταφέρονται σε διάρκεια δύο λεπτών είναι αρκετά μεγάλος. Πακέτα χάνονται κατά την προσπάθεια εξυπηρέτησης της κίνησης και λόγω του πρωτοκόλλου udp δεν υπάρχει επαναποστολή πακέτων.

Ο μέσος όρος λήψης πακέτων για τον h1 είναι 127.58 Mbps, h2 115,25 Mbps, h3 117,4 Mbps, h4 119 Mbps, h5 135.4 Mbps.

Στην παρακάτω εικόνα μπορούμε να δούμε τη λήψη πακέτων ανά host σε διάρκεια δύο λεπτών.



Κάθε host δέχεται και αποστέλλει τα πακέτα του από διαφορετική πόρτα του Openflow Switch 7. Κανένας host δεν χρησιμοποιεί την ίδια πόρτα του switch με κάποιον άλλο host. Το επόμενο switch, από το aggregation level, είναι το switch 6 και τελικό switch είναι το 1, το οποίο είναι ο εξυπηρετητής .

Στην παρακάτω εικόνα μπορούμε να δούμε, από το user interface του controller, το μονοπάτι που έχει επιλέξει ο controller.

1695388002	SDN2.00:00:00:00:00:00:00:01	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772162, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:01	Enqueue 0:2	null 32767
1135015621	SDN2.00:00:00:00:00:00:00:06	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772162, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:06	Enqueue 0:1	null 32767
1772059730	SDN2.00:00:00:00:00:00:00:07	ethernet-type=2048, protocol=6, ingress-port=-1, ip-dest=167772205, ip-src=167772162, tos-bits=34, vlan-id=-1, ethsrc=null, ethdst=null, tcpdstport=-1, tcpsrcport=-1,	00:00:00:00:00:00:00:07	Enqueue 0:4	null 32767

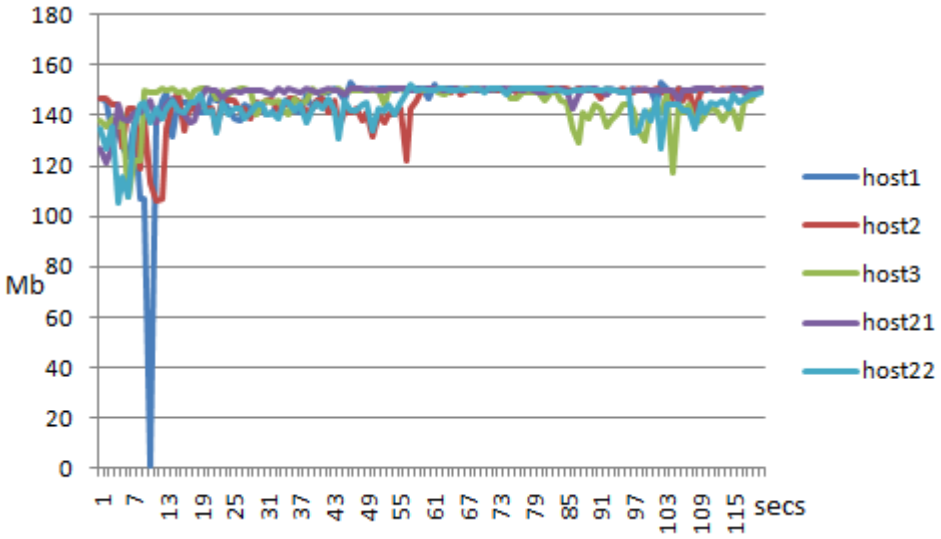
Στην εικόνα επίσης μπορούμε να δούμε και τις πολιτικές που ακολουθούνται σε αυτό το μονοπάτι. Οι πολιτικές που βλέπουμε εδώ έχουν περιγραφεί σε προηγούμενο κομμάτι του κεφαλαίου.

Στην επόμενη εικόνα μπορούμε να δούμε και ποιό switch με την αντίστοιχη πόρτα ανήκει στον host 1 (10.0.0.1) και host 3(10.0.0.3),αντιστοιχούν στα UE, καθώς και στο server(host 45, 10.0.0.45)

52:01:34:96:67:5f	10.0.0.1	00:00:00:00:00:00:00:07-5	4/5/2016, 7:33:31 π.μ.
d6:91:b4:4a:85:5c	10.0.0.3	00:00:00:00:00:00:00:07-7	4/5/2016, 7:33:31 π.μ.
36:c4:bc:19:d1:42	10.0.0.45	00:00:00:00:00:00:00:01-2	4/5/2016, 7:33:31 π.μ.

Στη περίπτωση που τρείς hosts (h1,h2,h3) οι οποίοι επικοινωνούν με το switch 7 ζητήσουν 150 Mbps video session από τον server h45, και ταυτόχρονα ζητούν από 150 Mbps video session και οι host 21 και 22, επικοινωνούν με το switch 11, από τον server h45, τότε το αποτέλεσμα είναι να λαμβάνουν οι host το εύρος ζώνης που ζητάνε.

Οι διαδρομές που επιλέγει ο controller διαφέρουν ανάλογα με τον τους host που ζητάνε πακέτα video. Όταν το request γίνεται από τους h1, h2, h3 επιλέγεται η διαδρομή switch7,switch6,switch1, ενώ όταν το request γίνεται από τους h21, h22 επιλέγεται η διαδρομή switch7,switch5,switch1.



Στην παραπάνω εικόνα μπορούμε να διακρίνουμε ότι λίγα πακέτα χάνονται .Η απόδοση είναι καλύτερη από την περίπτωση που και οι πέντε host βρίσκονται στο switch 7.

4.4 OpenEPC (Virtualized EPC)

Όσον αφορά το gtp-c και γενικότερα το control plane δεν μπορούμε να το εξομοιώσουμε με το mininet, ούτε να βγάλουμε κάποια αποτελέσματα με την χρήση αυτού του εργαλείου, καθώς το control plane θα βρίσκεται σε υποδομές στο cloud, το οποίο δεν μπορεί να υποστηριχτεί από το mininet.Το OpenEPC είναι ένα πρότζεκτ που εφαρμόζεται από το 3GPP EPC που αναπτύσσεται από την Fraunhofer FOKUS Competence Center για το Next Generation Network Infrastructure (NGNI) και το Technical University of Berlin (TU Berlin) [32]. Το OpenEPC αποτελείται από πολλά στοιχεία λογισμικού παρέχοντας εξελιγμένα σχεδιαγράμματα IP κινητικότητας, πολιτική βασισμένη στον έλεγχο του QoS, και ενσωμάτωση σε διαφορετικές πλατφόρμες εφαρμογών σε συγκλίνουσα δικτυακά περιβάλλοντα. Αυτό το εργαλείο μπορεί να χρησιμοποιηθεί για την εξομοίωση της S1-MME control plane interface. Τα πλεονεκτήματα ενός νέου control plane με όλα τα EPC στοιχεία να τρέχουν στη κορυφή ενός controller μπορούν να αποτελέσουν στοιχείο επόμενης εργασίας.

Μελλοντική Εξέλιξη

Σε αυτήν την εργασία είδαμε ένα μοντέλο αρχιτεκτονικής του EPC βασισμένο στο SDN, διαδικασίες load balancing σε αυτό το μοντέλο και εφαρμογή πολιτικών QoS. Σε κάθε περίπτωση οι αλγόριθοι που χρησιμοποιήθηκαν μπορούν να βελτιωθούν περαιτέρω, καθώς και η συντομότερη διαδρομή την οποία επιλέγει ο controller μπορεί και αυτή να τροποποιείται δυναμικά, ανάλογα με τις πληροφορίες για το φορτίο κίνησης που παρέχουν τα switches.

Επίσης, μπορεί να αποτελέσει αντικείμενο μελέτης η πλήρης εφαρμογή των EPC λειτουργιών ελέγχου στο cloud. Οι λειτουργίες αυτές μπορούν να εφαρμοστούν σαν ενσωματωμένες εφαρμογές π.χ. με το OpenStack ή σαν αυτόνομα στοιχεία. Ένα αυτόνομο MME θα περιλάμβανε το χτίσιμο της Representational State Transfer (REST) API διεπαφής μεταξύ MME και κεντρικού controller.

Επιπλέον, μια καλύτερη μελέτη για το πώς θα μπορούσαμε να συνδυάσουμε το NFV στην SDN αρχιτεκτονική μπορεί να υπάρξει.

Μια ερευνα αγοράς μπορεί να διενεργηθεί για να δούμε τις δυνατότητες κέρδους του προϊοντος. Το συμπέρασμα που πρέπει να καταλήξουμε είναι αν οικονομικά ευνοούνται οι πάροχοι από την επένδυση σε OpenFlow υλικό.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Korhonen, J. (2003). Introduction to 3G Mobile Communications. 2nd edition, Artech House

[2] Tara Ali-Yahiya (2011).Understanding LTE and Its Performance. Springer

[3] Halonen T., Romero J., Melero J., GSM, GPRS and EDGE Performance: Evolution Toward 3G/UMTS, Wiley, England, 2002

[4] THE EVOLUTION OF EDGE,February 2007,White Papers, ERICSSON

[5] Kaaranen H., Ahtiainen A., Laitinen L., Naghian S., Niemi V., *UMTS Networks: Architecture,Mobility and Services*, Wiley, England, 2005

[6] http://www.umts-forum.org/

[7] T. Halonen, J. Melero, J. R. Garcia, GSM,GPRS and EDGE Performance: Evolution Towards 3G UMTS, Halsted Press New York, NY, USA,2002

[8] Jiangzhou Wang,Tung-Sang NG, Advances in 3G Enhanced Technologies for Wireless , Artech House, 2002

[9] Haloma H., Toskala A., HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications, Wiley, England, 2006.

[10] ONF, "Software-Defined Networking: The New Norm for Networks," white paper, https://www.opennetworking.org

[11] Are We Ready for SDN? Implementation Challenges for Software-Defined Networks, Sakir Sezer, Sandra Scott-Hayward, and Pushpinder Kaur Chouhan, CSIT, Queen's University Belfast Barbara Fraser and David Lake, Cisco Systems, Jim Finnegan and Niel Viljoen, Netronome Marc Miller and Navneet Rao, Tabula,

[12] https://www.sdxcentral.com/resources/NFV/whats-network-functions-virtualization-NFV/

[13] https://portal.etsi.org/NFV/NFV_White_Paper.pdf ,Network Functions Virtualisation, White Paper, An Introduction, Benefits, Enablers, Challenges & Call for Action

[14] SDN and NFV Dynamic Operation of LTE EPC Gateways for Time-varying Traffic Patterns, SDN and NFV Dynamic Operation of LTE EPC Gateways for Time-varying Traffic Patterns, Arsany Basta, Andreas Blenk, Marco Homann, Hans Jochen Morper , Klaus Homann, and Wolfgang Kellerer

[15] Emulation of software defined networks using mininet in different simulation environments, Faris Keti,Shavan Askar

[16]A network in a laptop, rapid prototyping for software-defined networks, B.Lantz, B.Heller, N.McKeown , inProceedings of 9thACM SIGCOMM Workshop on Hot Topicsin Networks,ACM 2010

[17] Mininet An Instant Virtual Network on your Laptop, 2014,http://www.mininet.org

[18] NFV ISG PoC Proposal – virtual EPC with SDN Function in Mobile Backhaul Networks1,ETSI NFV PoC

[19] New Control Plane in 3GPP LTE/EPC Architecture for On-Demand Connectivity Service, Siwar Ben Hadj Said∗, Malla Reddy Sama∗, Karine Guillouard∗, Lucian Suciu∗ Gwendal Simon, Xavier Lagrange and Jean-Marie Bonnin

[20] Software Defined 5G Mobile Backhaul, Jose Costa-Requena, Raimo Kantola, Jesús Llorente,Vicent Ferrer, Jukka Manner, Aaron Yi Ding, Yanhe Liu, Sasu Tarkoma

[21] "TR 23.843 V12.0.0 3rd Generation Partnership Project;Technical Specification Group ServicesandSystemAspects;StudyonCoreNetworkOverload(CNO)solutions(Release 12)," December 2013.

[22] Ixia White Paper, "Quality of service (QoS) and policy management in mobile data networks – Validating Service Quality to ensure subscriber quality of experience (QoE)," 2013.

[23] NFV White Paper, "Network Functions Virtualization: an Introduction, Benefits, Challenges and Call for Action," 2012.

[24] Open Networking Foundation (ONF), "OpenFlow Switch Specification, version v.1.4.0," https://www.opennetworking.org/images/stories/downloads/SDN-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf, 2013,

[25] N. McKeown, "OpenFlow (Or: "Why can't I innovate in my wiring closet?")," http: //cleanslate.stanford.edu,

[26] 5g-what-is-it more than just improved performance and greater flexibility, the next generation is a shift of mindset, Ericsson, October 2014

[27] Προπτυχιακή εργασία στο μάθημα «Ευρυζωνικές Τεχνολογίες» με θέμα «Πέμπτης γενιάς κινητά δίκτυα επικοινωνιών ή Πέμπτη γενιά ασύρματων δικτύων (5G)»,Κόλλια Αναστασία, Πάτρα, Ιούνιος ,2014

[28]B. Liu, Software Defined Networking and Tunneling for Mobile Networks, 2013

[29]M. Howard, "Using Carrier Ethernet to Backhaul LTE, Infonetics Research," February, 2011.

[30] Ungureanu,Oana-Michaela "Flexible and Programmable Evolved Packet Core: A New SDN-based Model",Msc Thesis, 2014

[31] Open vSwitch, "Open Virtual Switch," http://openvswitch.org/, 2013

[32] "OpenEPC," http://www.openepc.net/index.html, 2012

[33] Li Erran Li,Morley Mao,Jennifer Rexford, "Toward Software-Defined Cellular Networks",2012

[34] Project Floodlight, "Floodlight," http://www.projectfloodlight.org/floodlight/, [Online; accessed 24-May-2014]

[35] "How to implement quality of service using floodlight," http://www.openflowhub. org/display/floodlightcontroller/How+to+implement+Quality+Of+Service+using+ Floodlight