ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«SDN και Ασφάλεια Δικτύων»

Της φοιτήτριας
Θέμα Αικατερίνης
Αρ. Μητρώου: 10/3677

Επιβλέπων καθηγητής
Ψαρράς Νικόλαος

Θεσσαλονίκη 2016

Περίληψη

Η έννοια «Δίκτυα Προσδιοριζόμενα από Λογισμικό» (Software Defined Networking) ή εν συντομία SDN αποτελεί μια νέα προσέγγιση στον σχεδιασμό, την κατασκευή και την διαχείριση των δικτύων. Η βασική ιδέα της έννοιας επικεντρώνεται στο διαχωρισμό του ελέγχου του δικτύου από το επίπεδο προώθησης, γεγονός που καθιστά ευκολότερη την βελτιστοποίηση του κάθε τομέα ξεχωριστά. Χαρακτηρίζεται ως μια από τις πλέον υποσχόμενες τεχνολογίες για να κάνουν τα δίκτυα προγραμματιζόμενα και δυνατά εικονοποίησης. Το SDN υπόσχεται να απλοποιήσει τη διαχείριση και τις λειτουργίες του δικτύου, να μειώσει το κόστος, να επιταχύνει την παροχή υπηρεσιών και να προωθήσει την καινοτομία μέσω του προγραμματισμού του δικτύου. Παρά το γεγονός ότι ο ανοικτός χαρακτήρας του SDN και η δυνατότητα προγραμματισμού είναι τα κύρια χαρακτηριστικά του, η ασφάλεια είναι βασικής σημασίας για την ανάπτυξη του.

Abstract

The term “Software Defined Networking” describes a new approach to the design, structure and management of networks. The basic idea of the concept is focused on separating network control from the forwarding plane, which makes it easier to optimize each segment separately. It is described as one of the most promising technologies to make networks programmable and virtualizeable. SDN promises to simplify management and network operations, reduce costs, accelerate service provisioning and promote innovation through network programming. Despite the fact that the openness of SDN and its programmability are its main features, security is of great importance for its development.

Ευχαριστίες

Η παρούσα πτυχιακή εργασία εκπονήθηκε κατά το ακαδημαϊκό έτος 2015-2016 στο Τμήμα Πληροφορικής του Αλεξάνδρειου Τεχνολογικού Εκπαιδευτικού Ιδρύματος Θεσσαλονίκης.

Θα ήθελα να ευχαριστήσω τον υπεύθυνο καθηγητή της πτυχιακής κ. Νικόλαο Ψαρρά για την ανάθεση της και τη δυνατότητα που μου έδωσε να ασχοληθώ με το συγκεκριμένο θέμα. Επίσης, όλους τους καθηγητές μου για όσα μου προσέφεραν όλα αυτά τα χρόνια.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου για τη στήριξη της καθ' όλη τη διάρκεια των σπουδών μου.

Θέμα Αικατερίνη

Ιούνιος 2016

Περιεχόμενα

Ευρετήριο σχημάτων

Ευρετήριο πινάκων

Εισαγωγή

Η έκρηξη των κινητών συσκευών, το εικονικό περιβάλλον των διακομιστών και η εμφάνιση των υπηρεσιών cloud έχουν οδηγήσει την βιομηχανία δικτύωσης να επανεξετάσει τις αρχιτεκτονικές δικτύου. Πίσω από όλες τις εταιρείες πολλών δισεκατομμυρίων δολαρίων, όπως η Google, Amazon, Facebook, βρίσκονται δίκτυα που πρέπει να προσαρμοστούν στις συνεχώς μεταβαλλόμενες ανάγκες, χωρίς να απαιτείται μεγάλη αλλαγή σε λογισμικό και υλικό. Οι παραδοσιακές αρχιτεκτονικές δικτύου θεωρούνται ακατάλληλες για να αντιμετωπίσουν τις σημερινές ανάγκες των υπηρεσιών, των διακομιστών και των χρηστών.

Ανανεώνοντας τους μεταγωγείς ώστε να διαχωριστούν τα επίπεδα ελέγχου και δεδομένων, ένας κεντρικός ελεγκτής δημιουργεί βελτιστοποιημένες διαδρομές για την προώθηση της κίνησης. Έτσι, αναπτύχθηκε μία νέα αρχιτεκτονική, η Software Defined Networking (SDN) η οποία επιτρέπει σε κέντρα δεδομένων και σε ερευνητές να καινοτομήσουν στα δίκτυα και να χειριστούν την δυναμική φύση των δικτύων υπολογιστών.

Όπως είναι γνωστό, ένα δίκτυο είναι η ασφάλεια του. Δίκτυα που υστερούν σε ασφάλεια χάνουν την ακεραιότητα τους. Το SDN προσφέρει ευκαιρίες για την ενίσχυση της ασφάλειας, οι περισσότερες από τις οποίες αναλύονται στα παρακάτω κεφάλαια.

Στο πρώτο κεφάλαιο αναλύεται η έννοια «Δίκτυα Προσδιοριζόμενα από Λογισμικό» καθώς και οι λόγοι που την κάνουν τόσο προσιτή. Στο δεύτερο κεφάλαιο παρουσιάζεται η αρχιτεκτονική του SDN και του πρωτοκόλλου OpenFlow ενώ στο τρίτο κεφάλαιο γίνεται επεξήγηση του τρόπου λειτουργίας του προτύπου. Στο τέταρτο κεφάλαιο αναφέρεται η έννοια της εικονοποίησης και η σχέση της με το SDN και στο πέμπτο κεφάλαιο περιγράφονται θέματα όπως τα πλεονεκτήματα, οι περιπτώσεις χρήσης καθώς και τα ζητήματα που ακόμη επικρατούν. Στη συνέχεια, στο έκτο κεφάλαιο, γίνεται μια εκτενής περιγραφή της ασφάλειας και πως οι υπηρεσίες αποδίδονται μέσα στο SDN ενώ στο έβδομο περιγράφεται ο προσομοιωτής δικτύων SDN, Mininet.

1. Τι είναι τα Δίκτυα Προσδιοριζόμενα από Λογισμικό

Το SDN επιτρέπει στις εφαρμογές να έχουν γνώση του δικτύου προσεγγίζοντας με έναν καινούριο τρόπο την αρχιτεκτονική των δικτύων. Ο καλύτερος ίσως τρόπος να καταλάβουμε το SDN είναι να το συγκρίνουμε με την συμπεριφορά των παραδοσιακών δικτύων. Σε ένα παραδοσιακό δίκτυο, ο δικτυακός εξοπλισμός, όπως ένας μεταγωγέας ή ένας δρομολογητής, περιέχουν το επίπεδο ελέγχου και το επίπεδο δεδομένων. Το επίπεδο ελέγχου περιγράφει την διαδρομή την οποία θα ακολουθήσουν τα πακέτα μέσα στο δίκτυο, ενώ το επίπεδο δεδομένων είναι το κομμάτι του δικτύου που φέρει τα πακέτα. Το SDN διαχωρίζει αυτές τις δύο λειτουργίες τοποθετώντας το επίπεδο ελέγχου, με άλλα λόγια την νοημοσύνη του δικτύου, κεντρικά σε βασιζόμενους σε λογισμικό ελεγκτές SDN, που διατηρούν μια γενική άποψη του δικτύου. Αυτό έχει ως αποτέλεσμα το δίκτυο να εμφανίζεται σε εφαρμογές ως ένας λογικός δρομολογητής. Με το SDN, οι διαχειριστές αποκτούν έλεγχο σε ολόκληρο το δίκτυο από ένα λογικό σημείο, κάτι το οποίο απλουστεύει σημαντικά τον σχεδιασμό του δικτύου και την λειτουργία του. Επίσης, το SDN απλουστεύει τις συσκευές δικτύου, καθώς δεν χρειάζεται πλέον να κατανοήσουν και να επεξεργαστούν χιλιάδες πρότυπα πρωτοκόλλου, αλλά να δεχτούν εντολές από τους ελεγκτές SDN.

Η Open Networking Foundation (ONF) είναι μια user-driven οργάνωση που έχει αναλάβει την ανάπτυξη και προτυποποίηση του SDN. Ο επίσημος ορισμός που δίνει η ONF καθορίζει το SDN ως «μια ανερχόμενη αρχιτεκτονική που είναι δυναμική, εύχρηστη, αποδοτική, και προσαρμόσιμη, καθιστώντας την ιδανική για τις υψηλού εύρους ζώνης σημερινές εφαρμογές. Η συγκεκριμένη αρχιτεκτονική διαχωρίζει τον έλεγχο του δικτύου από τις λειτουργίες προώθησης καθιστώντας τον έλεγχο του δικτύου άμεσα προγραμματίσιμο και τον υποκείμενο εξοπλισμό να αποσπαστεί από εφαρμογές και υπηρεσίες δικτύου. Το πρωτόκολλο OpenFlow είναι ένα θεμελιώδες στοιχείο για τις SDN λύσεις». Η αρχιτεκτονική SDN είναι:

- Άμεσα προγραμματίσιμη: Ο έλεγχος του δικτύου είναι άμεσα προγραμματιζόμενος επειδή είναι αποσυνδεδεμένος από τις λειτουργίες προώθησης.
- Ευέλικτη: Διαχωρίζοντας τον έλεγχο από την προώθηση δίνεται η δυνατότητα στους διαχειριστές δικτύων να προσαρμόζουν δυναμικά τη ροή της κίνησης σε όλο το δίκτυο ώστε ανταποκρίνεται σε μεταβαλλόμενες ανάγκες.
- Κεντρικά διαχειριζόμενη: Η νοημοσύνη του δικτύου είναι λογικά συγκεντρωμένη σε software-based ελεγκτές SDN που διατηρούν μια συνολική εικόνα του δικτύου.
- Προγραμματιστικά ρυθμιζόμενη: Το SDN επιτρέπει στους διαχειριστές δικτύου να ρυθμίσουν, να διαχειριστούν, να ασφαλίσουν καθώς και να βελτιστοποιήσουν τους πόρους του δικτύου πολύ γρήγορα μέσω δυναμικών, αυτοματοποιημένων προγράμματα SDN, τα οποία να μπορούν οι ίδιοι να γράψουν μιας και τα προγράμματα δεν εξαρτώνται από το ιδιόκτητο λογισμικό.

- Ουδέτερη από standards και προμηθευτές: Το SDN απλοποιεί το σχεδιασμό και τη λειτουργία του δικτύου επειδή οι οδηγίες που παρέχονται από τους ελεγκτές SDN και όχι από τον εκάστοτε προμηθευτή η πρωτόκολλο.

Η ONF είναι αφιερωμένη στην προώθηση και υιοθέτηση του SDN, στην εφαρμογή SDN μέσω ανοικτών προτύπων, τα οποία είναι απαραίτητα για να προχωρήσει η βιομηχανία δικτύωσης προς τα εμπρός. Ως μέρος της προσπάθειάς της να κάνει το SDN μια εμπορική πραγματικότητα που ανταποκρίνεται στις ανάγκες των πελατών, η ONF αναπτύσσει ανοικτά πρότυπα, όπως το *OpenFlow* και το *OpenFlow Configuration and Management Protocol Standard*. Το OpenFlow Standard είναι το πρώτο και μοναδικό ουδέτερο (από άποψη προμηθευτών) πρότυπο επικοινωνίας που ορίζεται μεταξύ των στρωμάτων ελέγχου και προώθηση μιας αρχιτεκτονικής SDN. Ομάδες εργασίας της ONF ανοίγουν το δρόμο για την ανάπτυξη διαλειτουργικών λύσεων μέσω της συνεργασίας με τους κορυφαίους ειδικούς στον κόσμο στην SDN και OpenFlow όσον αφορά έννοιες όπως frameworks, την αρχιτεκτονική και τα πρότυπα.

Το SDN έχει πολλές εφαρμογές συμπεριλαμβανομένων των Κέντρων Δεδομένων, των Δικτύων Κορμού (Backbone Networks), των Δικτύων τύπου Enterprise / Campus / Home και των Internet Excange Points (IXPs). Σε επόμενο κεφάλαιο θα αναλυθεί ο τρόπος με τον οποίο το SDN βοηθά να λυθούν προβλήματα διαχείρισης σε δίκτυα των παραπάνω κατηγοριών.

Για ποιο λόγο ο διαχωρισμός

Ο πρώτος λόγος για το διαχωρισμό του επιπέδου ελέγχου από το επίπεδο δεδομένων είναι οι ευκαιρίες που προσφέρει όσον αφορά στην ανεξάρτητη εξέλιξη και ανάπτυξη των δικτύων. Το λογισμικό και το υλικό δεν είναι πλέον αλληλένδετα και μπορούν να αναπτυχθούν ξεχωριστά. Ο δεύτερος λόγος που ο διαχωρισμός των δύο επιπέδων είναι αποδοτικός είναι η ευκαιρία να ελέγχεται η συμπεριφορά του δικτύου από ένα πρόγραμμα υψηλού επιπέδου, κάτι που επιτρέπει στους διαχειριστές ευκολότερο έλεγχο και αποσφαλμάτωση.

Προσφέρει επίσης καλύτερη διαχείριση δικτύου στα Κέντρα Δεδομένων με το να διευκολύνει την μετακίνηση των εικονικών μηχανών (Virtual Machine Migration) και περισσότερο έλεγχο στη λήψη αποφάσεων στις διαδικασίες προώθησης. Στα δίκτυα τύπου Enterprise μέσα από το SDN δίνεται η δυνατότητα να γραφούν εφαρμογές ασφαλείας, όπως εφαρμογές που διαχειρίζονται τον έλεγχο εισόδου στο δίκτυο (Network Access Control). Τέλος, στην Έρευνα ο διαχωρισμός επιτρέπει την εικονοποίηση ενός δικτύου ώστε τα ερευνητικά δίκτυα και τα πρωτόκολλα που ακόμα βρίσκονται σε πειραματικό στάδιο να συνυπάρχουν με δίκτυα παραγωγής στον ίδιο υποβόσκων εξοπλισμό.

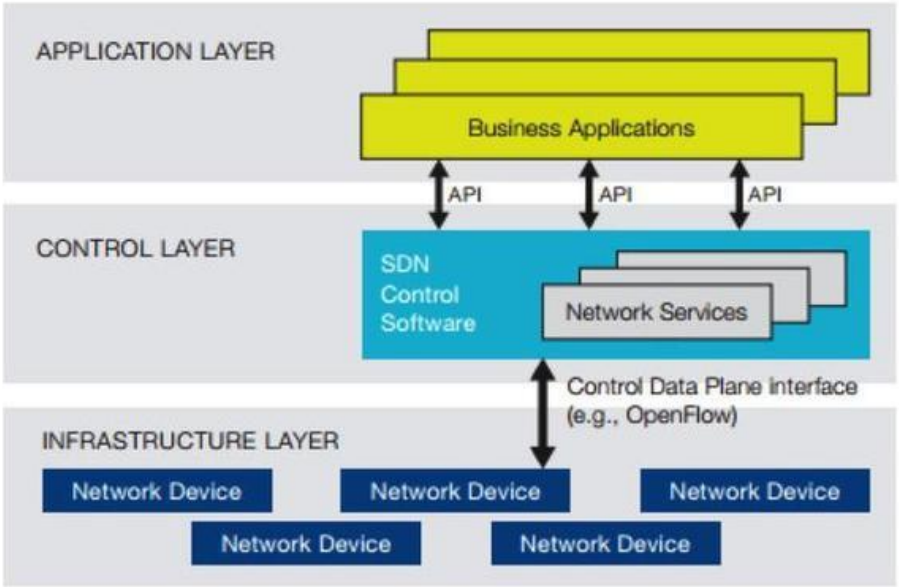
2. Η αρχιτεκτονική SDN

Η τεχνολογία του SDN διαχωρίζει το δίκτυο σε 3 επίπεδα: εφαρμογών, ελέγχου και δεδομένων ή υποδομών. (Εικόνα 1)

Το **επίπεδο εφαρμογών** περιέχει εφαρμογές SDN που επικοινωνούν μέσω διεπαφής προγραμματισμού εφαρμογών (Application Programming Interface, API). Οι εφαρμογές SDN είναι λογισμικά τα οποία επικοινωνούν απευθείας με τον ελεγκτή SDN για τις ανάγκες δικτύου και την επιθυμητή δικτυακή συμπεριφορά.

Το **επίπεδο ελέγχου**, ή ελεγκτής SDN, είναι ουσιαστικά ένα λειτουργικό σύστημα δικτύου γραμμένο σε μια γλώσσα υψηλού επιπέδου (Python, C, Java), το οποίο μεταβιβάζει τις ανάγκες από το επίπεδο εφαρμογών στο επίπεδο δεδομένων και παρουσιάζει έναν λογικό χάρτη του δικτύου στις εφαρμογές SDN που υλοποιούνται πάνω σε αυτό. Είναι η λογική στην οποία στηρίζονται οι κανόνες προώθησης. Για παράδειγμα, τα πρωτόκολλα δρομολόγησης και ο υπολογισμός της κατάλληλης διαδρομής που πρέπει να ακολουθήσει ένα πακέτο ανήκουν στο κομμάτι του επιπέδου ελέγχου. Οι αποφάσεις λαμβάνονται με μια γενική άποψη ολόκληρου του δικτύου και όχι με την περιορισμένη ορατότητα των γειτονικών δικτυακών συσκευών, όπως κάνουν οι δρομολογητές σήμερα.

Όταν αναφερόμαστε στο **επίπεδο δεδομένων** εννοούμε προγραμματίσιμο υλικό (hardware), ελεγχόμενο από το επίπεδο ελέγχου. Αφορά όλες εκείνες τις διαδικασίες που θα γίνουν, σύμφωνα πάντα με τους κανόνες που ορίζει το επίπεδο ελέγχου, για να προωθηθεί το εκάστοτε πακέτο στον τελικό του προορισμό. Ο Ελεγκτής επηρεάζει τις λειτουργίες που λαμβάνουν χώρα στο μεταγωγέα μέσα από εντολές ελέγχου. Το OpenFlow είναι το πρωτόκολλο που προσδιορίζει ένα σύνολο από εντολές ελέγχου μέσα από τις οποίες ο ελεγκτής μπορεί να ελέγχει τη συμπεριφορά ενός ή περισσότερων μεταγωγέων.



Εικόνα 1: Η αρχιτεκτονική SDN

2.1 OpenFlow

Το OpenFlow είναι ένα προγραμματιζόμενο πρωτόκολλο δικτύου και αποτελεί το πρώτο πρότυπο διεπαφής επικοινωνιών που ορίζεται μεταξύ των στρωμάτων ελέγχου και προώθησης σε μια αρχιτεκτονική SDN.

Το OpenFlow επιτρέπει την άμεση πρόσβαση και διαχείριση της κίνησης των δεδομένων του επιπέδου προώθησης καθώς και των συσκευών δικτύου (δρομολογητών, μεταγωγέων, επαναληπτών), εικονικά και φυσικά (hypervisor-based). Εφαρμόζεται μεταξύ των συσκευών υποδομής δικτύου και του λογισμικού ελέγχου SDN. Χρησιμοποιεί τους πίνακες ροής (flow tables) για την αναγνώριση της κίνησης δικτύου που βασίζεται σε κανόνες που έχουν προγραμματιστεί δυναμικά ή στατικά από το λογισμικό ελέγχου SDN. Επίσης, επιτρέπει την κατεύθυνση της κίνησης ορίζοντας παραμέτρους όπως μοτίβα χρήσης και εφαρμογές. Εφόσον στο πρωτόκολλο OpenFlow το δίκτυο προγραμματίζεται με βάση τις ροές, μια αρχιτεκτονική SDN – OpenFlow παρέχει εξαιρετικά λεπτομερή έλεγχο.

2.2 Northbound Interface

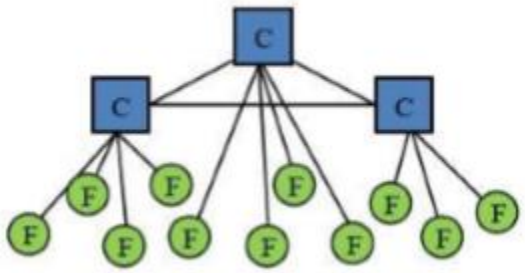
Με τον όρο «Βόρεια Διεπαφή» εννοείται μια διεπαφή που επιτρέπει σε ένα συγκεκριμένο δομικό στοιχείο ενός δικτύου να επικοινωνεί με ένα στοιχείο ανώτερου επιπέδου. Στην αρχιτεκτονική SDN οι βόρειες διεπαφές προγραμματισμού εφαρμογών (APIs) χρησιμοποιούνται στην επικοινωνία μεταξύ του ελεγκτή και του επιπέδου εφαρμογών.
Τα APIs είναι αναμφισβήτητα από τα πιο κρίσιμα στοιχεία στο περιβάλλον SDN, δεδομένου ότι η αξία του SDN συνδέεται με τις καινοτόμες εφαρμογές που μπορεί να υποστηρίξει. Επειδή είναι τόσο κρίσιμα πρέπει να υποστηρίζουν μια ευρεία ποικιλία εφαρμογών. Αυτός είναι πιθανώς ο λόγος που παραμένει σήμερα το πιο αόριστο συστατικό σε ένα περιβάλλον SDN μιας και υπάρχει μια ποικιλία πιθανών διεπαφών για τον έλεγχο διαφορετικού τύπου εφαρμογών μέσω ενός ελεγκτή SDN.

3. Το δίκτυο OpenFlow

Ένας μεταγωγέας είναι το βασικό δομικό στοιχείο ενός δικτύου OpenFlow: μπορεί να είναι ένας εμπορικός υλικός μεταγωγέας ή μια εικονική υλοποίηση. Μια εικονική υλοποίηση δρομολογητή είναι ο OpenVSwitch (OVS). Από την άλλη, το επίπεδο ελέγχου αποτελείται από τους Ελεγκτές OpenFlow (OpenFlow Controllers).

3.1 Ο ελεγκτής OpenFlow

Ο ελεγκτής αποτελεί μία βασική, κεντρική οντότητα η οποία εκτελεί όλες τις διεργασίες ελέγχου του OpenFlow δικτύου. Αυτό σημαίνει ότι παρέχει διεπαφή για την δημιουργία, την τροποποίηση και τον έλεγχο των πινάκων ροών των συσκευών που ελέγχει. Οι διαδικασίες προώθησης και δρομολόγησης δεδομένων επιτελούνται από τις συσκευές προώθησης βάση των καταχωρίσεων ροής (flow entries) όπως αυτές αποστέλλονται στην συσκευή από τον ελεγκτή. Οι ροές δεδομένων δημιουργούνται δηλαδή ανάλογα με την ζήτηση την κάθε χρονική στιγμή, ενώ ο ελεγκτής προσφέρει δυναμική ανάθεση πόρων. Αυτό προσφέρει σημαντική ευελιξία μιας και, όπως φαίνεται και στην εικόνα 2 είναι δυνατό να υπάρχουν πολλαπλοί OpenFlow ελεγκτές, οι οποίοι να επικοινωνούν αμφιμονοσήμαντα μεταξύ τους και ο κάθε controller να επικοινωνεί με τις δικές του συσκευές προώθησης πακέτων.



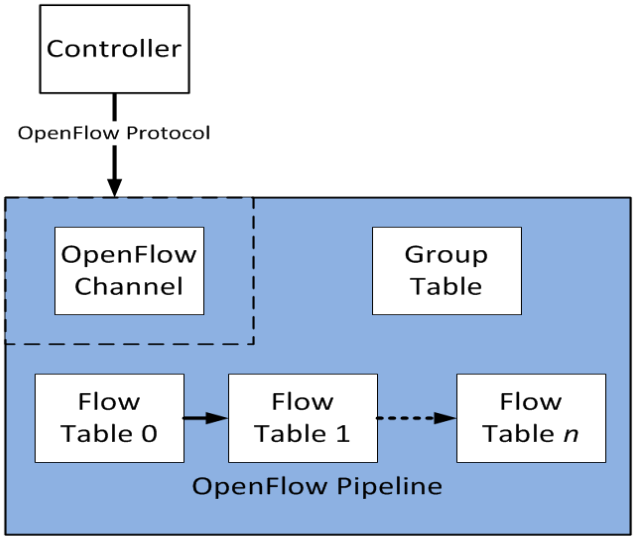
Εικόνα 2: Επικοινωνία ελεγκτών-μεταγωγέων

Υπάρχουν αρκετοί ελεγκτές OpenFlow ανοιχτού λογισμικού, σχεδόν για κάθε αναπτυξιακό περιβάλλον, όπως ο NOX σε C, οι POX, RYU σε Python, οι Floodlight, Beacon σε Java και πολλοί ακόμα.

3.2 Μεταγωγέας OpenFlow (OpenFlow Switch)

Στην εικόνα 3 φαίνεται ένας OpenFlow μεταγωγέας που επικοινωνεί με τον ελεγκτή μέσω ασφαλούς σύνδεσης χρησιμοποιώντας το πρωτόκολλο OpenFlow.

Ένας μεταγωγέας OpenFlow μπορεί να είναι οποιαδήποτε συσκευή προώθησης πακέτων που αποτελείται από έναν ή περισσότερους πίνακες ροής (flow tables) οι οποίοι χρησιμοποιούνται για την αναζήτηση και προώθηση πακέτων καθώς και έναν



Εικόνα 3: Δομή του OpenFlow Switch

πίνακα ομαδοποίησης (group table), ο οποίος εκτελεί την ανίχνευση πακέτων και προώθηση. Τρίτο συστατικό του μεταγωγέα είναι ένα κανάλι OpenFlow για άμεση επικοινωνία με έναν εξωτερικό ελεγκτή. Μέσω του καναλιού αυτού ο μεταγωγέας επικοινωνεί με τον ελεγκτή και ο ελεγκτής διαχειρίζεται τον μεταγωγέα.

Χρησιμοποιώντας το πρωτόκολλο OpenFlow, ο ελεγκτής μπορεί να προσθέσει, να ανανεώσει ή να διαγράψει καταχωρήσεις ροής (flow entries) στους πίνακες ροής, διαδραστικά (σαν απάντηση σε αιτήσεις πακέτων) ή προληπτικά.

Κάθε πίνακας ροής περιέχει μια συλλογή καταχωρήσεων ροής, οι οποίες αποτελούνται από πεδία αντιστοιχίας (match fields), μετρητές (counters), και μια συλλογή οδηγιών (instructions), για να εφαρμόσουν στα αντιστοιχισμένα πακέτα.

Η αντιστοίχιση ξεκινά από τον πρώτο πίνακα ροής και μπορεί να συνεχίσει και στους υπόλοιπους, αν υπάρχουν, αφού οι πίνακες ροής βρίσκονται σε διασωληνωμένη μορφή (pipelined). Στην περίπτωση που βρεθεί καταχώρηση που να ταιριάζει με το πακέτο, τότε εκτελούνται οι οδηγίες που συνοδεύουν την συγκεκριμένη καταχώρηση. Εάν δεν βρεθεί καμία αντιστοιχία στον πίνακα ροής, το αποτέλεσμα εξαρτάται από τη διαμόρφωση του μεταγωγέα: το πακέτο μπορεί να διαβιβάζεται στον ελεγκτή μέσω του καναλιού OpenFlow, να απορριφθεί, ή να συνεχίσει στον επόμενο πίνακα ροής.

Οδηγίες που σχετίζονται με την εκάστοτε καταχώρηση, περιγράφουν την προώθηση και την τροποποίηση των πακέτων, την επεξεργασία του πίνακα ομάδας (group table) , και την διαδικασία της διοχέτευσης. Οι οδηγίες που αναφέρονται στην διεργασία της διοχέτευσης, επιτρέπουν στα πακέτα να αποστέλλονται σε επόμενους πίνακες για

περαιτέρω επεξεργασία καθώς επίσης και την ανταλλαγή πληροφοριών, με τη μορφή μεταδεδομένων, μεταξύ των πινάκων. Η διαδικασία σταματά όταν το σετ οδηγιών που σχετίζεται με ένα flow entry δεν προσδιορίζει έναν επόμενο πίνακα. Σε αυτό το σημείο το πακέτο συνήθως τροποποιείται και προωθείται.

Οι καταχωρήσεις ροής προωθούνται σε μια θύρα η οποία μπορεί να είναι φυσική ή/και εικονική. Οι εικονικές θύρες μπορούν να εκτελέσουν διαδικασίες προώθησης, όπως αποστολή στον ελεγκτή ή υπερχείλισης χρησιμοποιώντας μεθόδους χωρίς την χρήση OpenFlow, όπως δηλαδή η συμβατική λειτουργία ενός μεταγωγέα.

Πέραν της διαχείρισης κάθε πακέτου ξεχωριστά, οι μεταγωγείς μπορούν να επεξεργαστούν μαζικά την κίνηση με την χρήση πινάκων ομάδας (group tables). Ένα flow entry μπορεί να αντιστοιχεί σε ένα group table action (ενέργεια που ορίζεται στο group table) για πακέτα που αντιστοιχούν στο εν λόγω flow entry. Τέτοιες ενέργειες που επιτελούνται βάση και με χρήση του group table και των group entries που αυτό περιλαμβάνει, μπορούν να είναι πιο πολύπλοκες, όπως η πολυδιόδευση πακέτων (multipath forwarding), γρήγορη επαναδρομολόγηση (fast reroute) ή η συσσωμάτωση ζεύξεων (link aggregation).

Οι group tables περιέχουν group entries. Κάθε μια από αυτές περιέχει μια λίστα από action buckets με συγκεκριμένη σημασιολογία που εξαρτάται από τον τύπο της εκάστοτε ομάδας. Οι δράσεις που περιέχονται σε κάθε action bucket εφαρμόζονται σε πακέτα που στέλνονται στην ομάδα αυτή.

Όπως αναφέρθηκε και προηγουμένως τα flow tables σε ένα μεταγωγέα βρίσκονται σε διασωληνωμένη μορφή (pipelined) και ανάλογα με την μορφή της διασωλήνωσης οι μεταγωγείς διακρίνονται σε δύο είδη:

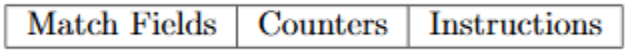
- Οι OpenFlow-only μεταγωγείς, οι οποίοι υποστηρίζουν μόνο λειτουργίες OpenFlow και όλα τα πακέτα επεξεργάζονται με την OF διασωλήνωση του μεταγωγέα.

- Οι OpenFlow-hybrid μεταγωγείς, οι οποίοι υποστηρίζουν και λειτουργία OpenFlow καθώς και λειτουργία Ethernet (κλασσική λειτουργία OSI layer 2 και 3 δρομολόγησης). Αυτοί οι μεταγωγείς περιλαμβάνουν ένα μηχανισμό κατηγοριοποίησης που ανακατευθύνει τα ληφθέντα πακέτα είτε στην διασωλήνωση OF είτε στη κανονική διασωλήνωση Ethernet. Ο μηχανισμός αυτός μπορεί να βασίζεται στην θύρα εισόδου των πακέτων ή πιθανά σε κάποια ετικέτα (tag) στην επικεφαλίδα (header) των πακέτων.

Οι σχεδιαστές των μεταγωγέων μπορούν να υλοποιήσουν τα εσωτερικά κατασκευαστικά στοιχεία με όποιον τρόπο επιθυμούν, με τον όρο οι συσκευές να πληρούν την βασική αρχιτεκτονική που απαιτείται για το πρωτόκολλο OpenFlow όπως αυτή αναλύθηκε προηγουμένως. Φυσικά μπορούν να γίνουν διαφοροποιήσεις που να είναι συμβατές με το OF, για παράδειγμα η χρήση μιας μάσκας bit (bitmask) για προώθηση πακέτων σε πολλαπλές θύρες σε ένα flow entry, αντί να γίνει χρήση του group table.

3.2.1 Flow Table

Η δομή ενός πίνακα ροών αποτελείται από τα αντίστοιχα flow entries όπως φαίνεται και στην Εικόνα 4:

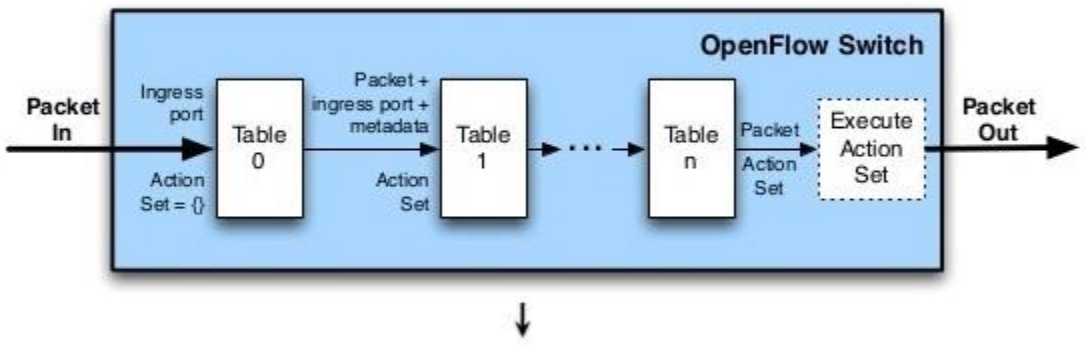


Εικόνα 4: Δομή Flow Entry

Βάση αυτών των καταχωρίσεων γίνεται η δρομολόγηση των πακέτων από το πρωτόκολλο OpenFlow. Αναλυτικότερα η δομή ενός flow entry είναι:

- Match fields: τα πεδία αντιστοίχισης για τα ληφθέντα πακέτα. Αυτά περιέχουν την θύρα εισόδου (ingress port), τις επικεφαλίδες και θύρες πακέτων (headers, ports) και προαιρετικά κάποια μεταδεδομένα (metadata).
- Counters: μετρητές που αφορούν τα ληφθέντα πακέτα, για παράδειγμα πόσα αντιστοιχήθηκαν επιτυχώς ή πόσα απορρίφθηκαν.
- Instructions: οδηγίες για την επιτέλεση ή τροποποίηση κάποιων ενεργειών που γίνονται κατά την διάρκεια της επεξεργασίας του πακέτου.

Για την διαδοχική σύγκριση στα flow tables το OpenFlow ακολουθεί την διαδικασία της διοχέτευσης (pipeline) δημιουργώντας έτσι μια σταθερή ροή πακέτων μέσω των flow tables, όπως μπορεί κανείς να διαπιστώσει και από το ακόλουθο σχήμα:



Εικόνα 5: Διαδρομή πακέτου στο OF Switch

Όπως γίνεται αντιληπτό και από το σχήμα οι πίνακες ροής ενός OpenFlow δρομολογητή είναι αριθμημένοι ακολουθιακά, ξεκινώντας από το 0. Η επεξεργασία αρχίζει σειριακά. Η διαδικασία της διοχέτευσης ξεκινάει πάντα από τον πρώτο πίνακα ροής. Για να γίνει η αντιστοίχιση κατά την λήψη ενός πακέτου γίνεται σύγκριση των δεδομένων της

επικεφαλίδας του πακέτου, της θύρας εισόδου του πακέτου και τυχόν μεταδεδομένων διαδοχικά με το πρώτο flow entry του πρώτου flow table και σταδιακά όλων των επόμενων entries στα υπόλοιπα flow tables. Η αντιστοίχιση γίνεται κατά σειρά προτεραιότητας, δηλαδή η πρώτη επιτυχής αντιστοίχιση flow entry είναι αυτή που ακολουθείται, με τα επόμενα entries στα υπόλοιπα flow tables να χρησιμοποιούνται ανάλογα με τις οδηγίες του matched entry.

Όταν επιτύχει η αντιστοίχιση ενός πακέτου με ένα flow entry ενός flow table, εκτελείται το αντίστοιχο σετ εντολών (Instructions set) που περιλαμβάνει το entry. Οι εντολές αυτές μπορεί να ανακατευθύνουν το πακέτο σε άλλο flow table όπου η διαδικασία προώθησης επαναλαμβάνεται (εντολές Goto). Ένα flow entry μπορεί να κατευθύνει το πακέτο σε flow table με αριθμό μεγαλύτερο από το τρέχον (στο οποίο βρίσκεται το entry), καθώς η διαδικασία της διοχέτευσης λειτουργεί με προώθηση του πακέτου προς τα εμπρός και όχι προς τα πίσω. Όπως είναι προφανές, τα flow entries του τελευταίου πίνακα ροής δεν μπορούν να περιλαμβάνουν εντολές που αφορούν την διαδικασία προώθησης (εντολές Goto). Όταν το πακέτο φτάσει στο τέλος της διασωλήνωσης και δεν υπάρχει άλλη εντολή προώθησης τότε το πακέτο επεξεργάζεται από τις εντολές του flow entry και προωθείται στην ανάλογη θύρα εξόδου.

Στην περίπτωση που ένα πακέτο δεν αντιστοιχηθεί σε κάποιο flow entry ενός flow table τότε έχουμε αστοχία πίνακα – table miss. Η διαδικασία που ακολουθείται σε τέτοιες περιπτώσεις εξαρτάται από την πολιτική του εκάστοτε δικτύου OpenFlow. Ως προεπιλογή στην περίπτωση αυτή τα αντίστοιχα πακέτα προωθούνται μέσω του καναλιού ελέγχου (control channel) στον ελεγκτή. Βέβαια σε ορισμένες περιπτώσεις ακολουθείται και η διαδικασία απόρριψης του πακέτου (drop).

3.2.2 Group Table

Κάθε καταχώρηση ομάδας (group entry) περιέχει:

- Group Identifier: αναγνωριστικό ομάδας, που είναι ένας μη προσημασμένος ακέραιος αριθμός των 32bit και χρησιμεύει για την ταυτοποίηση της ομάδας πακέτων που αφορούν το συγκεκριμένο entry.

- Group Type: ο τύπος της ομάδας, που δείχνει ποιες εντολές από την ομάδα εντολών (action bucket) θα εκτελεστούν για τα πακέτα που ανήκουν στο entry.

- Counters: μετρητές, που χρησιμοποιούνται για διάφορες πληροφορίες, όπως αριθμός πακέτων που ανήκουν στο group entry.

- Action Buckets: Διατεταγμένη λίστα εντολών στην οποία κάθε εντολή περιέχει ένα σύνολο ενεργειών και τις σχετικές παραμέτρους για να εφαρμοστούν στο προς επεξεργασία πακέτο.

Group Identifier	Group Type	Counters	Action Buckets

Εικόνα 6: Δομή Group Entry

Όσον αφορά τα group types, υπάρχουν τέσσερις κατηγορίες:

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

- Κατηγορία select: Εκτελείται μόνο μια εντολή από το σύνολο της ομάδας εντολών, βάσει των παραμέτρων του πακέτου και των αλγορίθμων επιλογής της εκάστοτε συσκευής.

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

- Κατηγορία fast failover: Εκτελείται η πρώτη εν ενεργεία ομάδα εντολών, δηλαδή το σύνολο εντολών που είναι συσχετισμένο με μια ενεργή θύρα εξόδου. Με αυτή την μέθοδο κάθε δέσμη ενεργειών στο action bucket συσχετίζεται με μια θύρα εξόδου για να μπορεί η συσκευή δρομολόγησης σε περίπτωση απώλειας σύνδεσης από μια θύρα εξόδου να επαναδρομολογήσει πακέτα σε εφεδρικές θύρες χωρίς να χρειαστεί ξανά επικοινωνία με τον controller.

3.2.3 Ασφαλές Κανάλι (Secure Channel)

Το ασφαλές κανάλι (secure channel) είναι η διεπαφή που συνδέει κάθε μεταγωγέα OpenFlow σε έναν ελεγκτή. Μέσω της διεπαφής αυτής ο ελεγκτής ανταλλάσσει μηνύματα με τις δικτυακές συσκευές προκειμένου να τις ρυθμίσει και να τις διαχειριστεί.
Το OpenFlow παρέχει ένα πρωτόκολλο για την επικοινωνία μεταξύ του ελεγκτή και τους OpenFlow-μεταγωγείς. Υπάρχουν τρεις τύποι μηνυμάτων που υποστηρίζονται από το πρωτόκολλο OpenFlow. Τα μηνύματα τύπου controller-to-switch, asynchronous και symmetric.

- Τα μηνύματα τύπου Controller-to-switch αποστέλλονται από τον ελεγκτή και δεν απαιτούν πάντα απάντηση από τον δρομολογητή. Μέσα από αυτά τα μηνύματα ο ελεγκτής ρυθμίζει τις δικτυακές συσκευές, διαχειρίζεται τον πίνακα ροής και αποκτά πληροφορίες για την κατάσταση του και τις δυνατότητες που υποστηρίζονται από τον δρομολογητή κάθε δεδομένη στιγμή.

- Τα ασύγχρονα μηνύματα αποστέλλονται χωρίς αίτηση από τον μεταγωγέα στον ελεγκτή και δηλώνουν μια αλλαγή στο μεταγωγέα ή στην κατάσταση του δικτύου. Ένα από τα πιο σημαντικά γεγονότα είναι το packet-in, το οποίο συμβαίνει όταν ένα πακέτο που δεν έχει αντιστοίχιση με κάποια ροή, φτάνει στον δρομολογητή. Όταν συμβαίνει αυτό, ένα μήνυμα packet-in στέλνεται στον ελεγκτή, που περιέχει το πακέτο ή ένα μέρος του πακέτου, προκειμένου για τον ελεγκτή να εξετάσει και

να προσδιορίσει ποιο είδος ροής θα πρέπει να συσταθεί για αυτό. Άλλα γεγονότα περιλαμβάνουν λήξη καταχωρήσεων ροής, αλλαγή κατάστασης της θύρας ή άλλα συμβάντα σφάλματος.

- Τέλος, η τρίτη κατηγορία των μηνυμάτων OpenFlow είναι τα συμμετρικά που αποστέλλονται χωρίς αίτηση προς τις δύο κατευθύνσεις. Τα μηνύματα αυτά μπορούν να χρησιμοποιηθούν για να βοηθήσουν ή να διαγνώσουν προβλήματα στην σύνδεση του ελεγκτή-μεταγωγέα.

4. Εικονοποίηση Δικτύου (Network Virtulization)

Η εικονοποίηση ενός δικτύου είναι μια αφαίρεση (abstraction) του φυσικού δικτύου όπου πολλά λογικά δίκτυα μπορούν να μοιράζονται τον ίδιο φυσικό εξοπλισμό. Ένας από τους κύριους λόγους που η βιομηχανία στράφηκε στην εικονοποίηση των δικτύων είναι η αποστέωση (ossification) της αρχιτεκτονικής του διαδικτύου. Συγκεκριμένα, το πρωτόκολλο δικτύου ήταν τόσο «διάχυτο» (pervasive) που έκανε πολύ δύσκολη την πραγματοποίηση θεμελιωδών αλλαγών στον τρόπο που η ήδη υπάρχουσα αρχιτεκτονική λειτουργούσε. Αντί λοιπόν να αλλαχθεί ο προϋπάρχων τρόπος λειτουργίας προτιμήθηκε η εικονοποίηση ώστε να επιτραπεί η εξέλιξη. Με άλλα λόγια, η εικονοποίηση των δικτύων επιτρέπει την εξέλιξη γιατί επιτρέπει σε διαφορετικές αρχιτεκτονικές να συνυπάρχουν παράλληλα.

Η εικονοποίηση των δικτύων έχει πολλές προοπτικές μεταξύ των οποίων είναι και η ταχεία εξέλιξη, αφού πλέον οι ρυθμοί ορίζονται σε ταχύτητες λογισμικού και όχι υλικού, οι νέες μορφές ελέγχου του δικτύου αλλά και ο απλούστερος προγραμματισμός.

4.1 Τι είναι η Εικονοποίηση Λειτουργιών Δικτύου

Το NFV (Network Functions Virtualization) έχει να κάνει με την μετάβαση των δικτύων από συλλογές ιδιωτικών κουτιών σε συλλογές στοιχείων λογισμικού που τρέχουν σε industry-standard υλικό. Στα παραδοσιακά δίκτυα, κάθε ξεχωριστή λειτουργία εφαρμόζεται σαν μια ξεχωριστή συσκευή, για παράδειγμα, δρομολογητές, τείχη προστασίας, load balancers κλπ.
Το NFV βασίζεται στην πρόταση οι λειτουργίες αυτές να εφαρμόζονται και να διατίθενται στο δίκτυο σαν ένα κομμάτι λογισμικού το οποίο εγκαθιστά ο χειριστής του δικτύου σε τυποποιημένες υποδομές υλικού (standardized hardware infrastructure).

4.2 Η σχέση μεταξύ SDN και NFV

Οι Network Functions Virtualization (NFV) και Software Defined Networking (SDN) θεωρούνται από πολλούς ως οι δύο τεχνολογίες που θα διαμορφώσουν το μέλλον των τηλεπικοινωνιακών δικτύων:

- Η τεχνολογία NFV, αποσκοπεί στην μεταφορά δικτυακών ή τηλεπικοινωνιακών εφαρμογών, που σήμερα συνήθως λειτουργούν σε αποκλειστικές και εξειδικευμένες πλατφόρμες, σε εικονικές υποδομές cloud.

- Η τεχνολογία SDN αποσκοπεί στην αύξηση της ικανότητας του δικτύου να προσαρμόζεται δυναμικά στις ανάγκες των εφαρμογών και υπηρεσιών που εξυπηρετεί.

Οι δύο τεχνολογίες έχουν πολλά κοινά. Η κεντρική ιδέα είναι ο διαχωρισμός του υλικού από το λογισμικό σε ένα δίκτυο και η αξιοποίηση χαμηλού κόστους εξοπλισμού με

ανεξάρτητα ανεπτυγμένο λογισμικό. Επίσης και οι δύο τεχνολογίες προβλέπουν υψηλό βαθμό αυτοματοποίησης και διαχείρισης δικτυακών υπηρεσιών, κάτι που επιτυγχάνεται με το να «τρέχει» NFV και SDN λογισμικό σε cloud περιβάλλον. Το σημείο που το SDN υπερβαίνει το NFV είναι ότι εισάγει την δυνατότητα εξόδου από του περιορισμούς που επιβάλλονται στα παραδοσιακά πρωτόκολλα δρομολόγησης επιτρέποντας έτσι την διαδικασία της δρομολόγησης και βελτιστοποίησης της κίνησης να εκτελεστούν με νέους τρόπους.

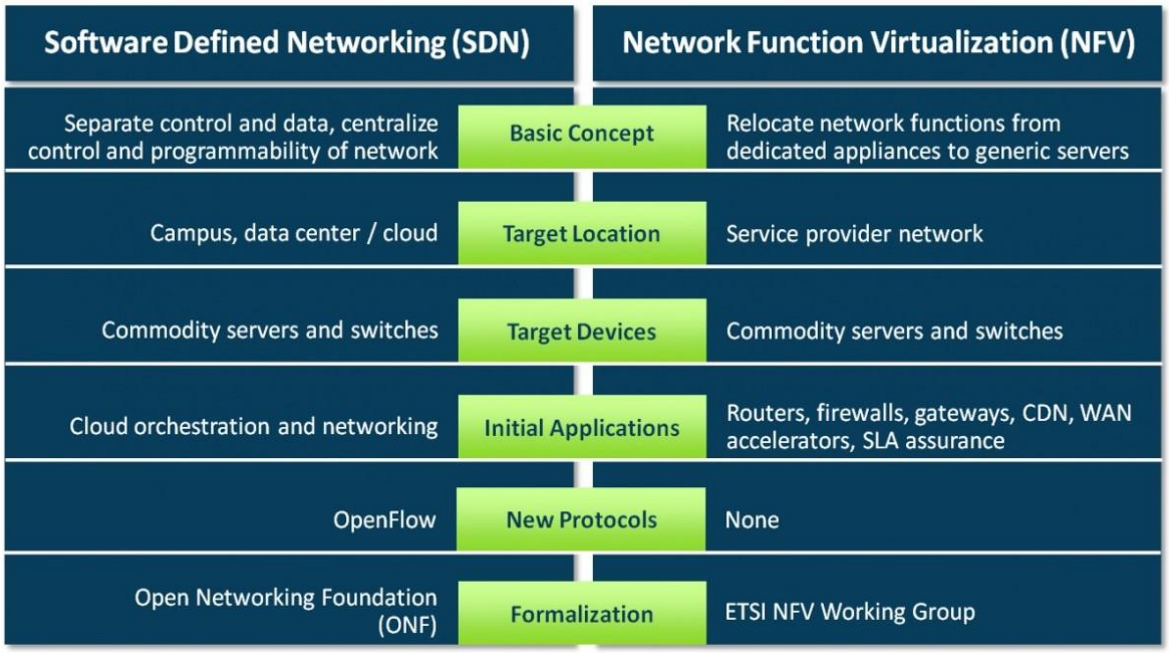
Υπάρχουν τέσσερεις γενικοί παράγοντες που απαρτίζουν μια ολοκληρωμένη πλατφόρμα NFV:

1. Ένα διαχειριζόμενων ροών δίκτυο μπορεί να προγραμματιστεί προβλέψιμα και με ασφαλή τρόπο ώστε τα πακέτα, οι ροές και οι διαδρομές να μπορούν να ενσωματωθούν με κινητές εικονικές συσκευές.

2. SDN εφαρμογές και ελεγκτής που μπορούν να διαχειριστούν, να παρακολουθήσουν και να αναλύσουν την κατάσταση του συνολικού δικτύου και να ενσωματωθούν με το σύστημα του NFV ελεγκτή.

3. Οι υπηρεσίες δικτύου είναι βασικές λειτουργίες ενός δικτύου, όπως ένα τείχος προστασίας. Ο παράγοντας-κλειδί είναι πως λειτουργούν ρε ροές πρωτοκόλλων μεταξύ των άκρων του δικτύου.

4. Υπηρεσίες χρηστών που υποστηρίζουν υπηρεσίες δικτύου ή παρέχουν επιπλέον λειτουργίες.



Εικόνα 7: Σύγκριση SDN και NFV

5. Πλεονεκτήματα – Θέματα – Περιπτώσεις Χρήσης

5.1 Πλεονεκτήματα του SDN από τα παραδοσιακά δίκτυα

Η νέα αρχιτεκτονική που συστήνει το SDN προσφέρει μια σειρά από πλεονεκτήματα έναντι των παραδοσιακών δικτύων με το σημαντικότερο να είναι η ευελιξία στο πως το δίκτυο χρησιμοποιείται και λειτουργεί. Το λογισμικό που το διέπει μπορεί να γραφτεί από επιχειρήσεις και παρόχους υπηρεσιών χρησιμοποιώντας συνηθισμένα περιβάλλοντα λογισμικού.

Ένα άλλο πλεονέκτημα είναι και η ορατότητα των πόρων του δικτύου. Μέχρι σήμερα οι εφαρμογές που «τρέχουν» στο επίπεδο εφαρμογών του μοντέλου Open Systems Interconnection (OSI) έχουν μικρή ή και καθόλου ορατότητα στους πόρους του δικτύου. Στο SDN ο ελεγκτής μπορεί, αφού δει τους απαιτούμενους πόρους των εφαρμογών και τη διαθεσιμότητα των πόρων στα κατώτερα στρώματα, να ταιριάξει αυτές τις δύο έννοιες, με αποτέλεσμα την αποδοτικότερη χρήση του υλικού δικτύωσης.

Επίσης, ενώ έως σήμερα η νοημοσύνη του δικτύου (π.χ. οι αλγόριθμοι για να βρεθεί η κατάλληλη διαδρομή) και η λήψη αποφάσεων διαμοιράζονται μέσα στο δίκτυο, στο SDN βρίσκονται στον ελεγκτή, κάτι που μειώνει δραστικά την πολυπλοκότητα των στοιχείων του δικτύου.

Άλλο ένα μεγάλο πλεονέκτημα είναι η προγραμματιστικότητα του ελεγκτή από γλώσσες υψηλού επιπέδου σε αντίθεση με τη ρύθμισης κάθε συσκευής ξεχωριστά από το Command Line Interface (CLI) της και τη δυσκολία της διαδικασίας αποσφαλμάτωσης.

Η απλούστευση του υλικού δικτύωσης αναμένεται να προκαλέσει αύξηση του ανταγωνισμού που οδηγεί σε μείωση του κόστους εξοπλισμού. Λόγω της ορατότητας του δικτύου που προσφέρεται από το SDN, το υλικό του δικτύου που ήταν συνήθως περισσότερο από το απαιτούμενο μπορεί να καθοριστεί με μεγαλύτερη ακρίβεια κάτι που αναμένεται να μειώσει περαιτέρω τις κεφαλαιακές δαπάνες.

Η προγραμματιζόμενη φύση της αρχιτεκτονικής SDN καθιστά ευκολότερο το σχεδιασμό, την ανάπτυξη, τη διαχείριση και την κλιμάκωση των δικτύων. Η ικανότητα της αυτοματοποίησης των προβλέψεων προσφέρει ευελιξία στα δίκτυα και μειώνει τις απαιτήσεις του ανθρώπινου δυναμικού, με αποτέλεσμα χαμηλότερες λειτουργικές δαπάνες.

5.2 Περιπτώσεις χρήσης

Υπάρχει πληθώρα περιπτώσεων χρήσης όπου και οι δύο τεχνολογίες (SDN, NFV) προσφέρουν οφέλη στις επιχειρήσεις που προσπαθούν να μετακινηθούν σε ένα εικονικό περιβάλλον καθώς επίσης και λύση στα σημερινά προβλήματα δικτύωσης που προκύπτουν.

Για φορείς και παρόχους υπηρεσιών προσφέρουν εύρος ζώνης κατά ζήτηση (on demand) ενώ στο cloud και στα data centers η εικονοποίηση προσφέρει οφέλη στην αξιοποίηση πόρων. Σε δίκτυα τύπου Enterprise ή Campus αξιοποιείται καλύτερα ο έλεγχος πρόσβασης στο δίκτυο (Network Access Control) καθώς και η ποιότητα υπηρεσιών (Quality of Service). Όλα αυτά περιγράφονται αναλυτικότερα στις παρακάτω παραγράφους. Σε κάποια παραδείγματα οι περιπτώσεις χρήσης είναι γενικές ενώ κάποια άλλα αποτελούν πραγματικές υλοποιήσεις.

Virtual Machine Mitigation

Ένα από τα πλεονεκτήματα της εικονοποίησης των servers είναι πως επιτρέπει την μεταφορά εικονικών μηχανών μεταξύ φυσικών servers. Ωστόσο, όταν μια εικονική μηχανή μετακινείται μεταξύ servers χρειάζεται να είναι στο ίδιο VLAN πριν και μετά τη μετακίνηση. Η επέκταση των VLANs σε ένα data center ώστε να υποστηρίζονται οι μετακινήσεις αυτές προσθέτει λειτουργικό κόστος και περιπλοκότητα καθώς και χρόνο επεξεργασίας επειδή απαιτεί κάθε μεταγωγέας κατά μήκος της διαδρομής να επαναρυθμιστεί χειροκίνητα.

Η εικονοποίηση των δικτύων επιλύει αυτό το πρόβλημα διότι ακόμα και με την μετακίνηση μιας εικονικής μηχανής σε ένα διαφορετικό υποδίκτυο στο ίδιο φυσικό δίκτυο, οι μεταγωγείς στην άκρη του overlay δικτύου ανανεώνουν αυτόματα τους πίνακες χαρτογράφησης (mapping tables) ώστε να αντικατοπτρίζουν τη νέα φυσική τοποθεσία της εικονικής μηχανής. Ένα από τα πλεονεκτήματα της εικονοποίησης δικτύων είναι πως αφού οι απαραίτητες αλλαγές πραγματοποιούνται μόνο στην άκρη του δικτύου δεν χρειάζεται να γίνει τίποτα στο υπόλοιπο δίκτυο.

Αλυσιδωτή σύνδεση υπηρεσιών (Service Chaining)

Σε ένα παραδοσιακό data center η εφαρμογή υπηρεσιών των επιπέδων 4-7, όπως ένα τείχος προστασίας, είναι επίπονη και χρονοβόρα μιας και απαιτεί την απόκτηση των απαιτούμενων συσκευών δικτύου και την καλωδίωση μεταξύ τους στη σωστή σειρά. Μιας και κάθε συσκευή έχει το δικό της ξεχωριστό σημείο σύνδεσης (interface), η ρύθμιση όλων των παραμέτρων κάθε συσκευής ξεχωριστά είναι μια επίσης χρονοβόρα καθώς και επιρρεπής σε λάθη διαδικασία.

Το SDN ξεπερνά αυτές τις προκλήσεις με την εφαρμογή δυο στενά συνδεδεμένων εφαρμογών: *Εισαγωγή Υπηρεσιών* (Service Insertion) και *Αλυσιδωτή Σύνδεση Υπηρεσιών* (Service Chaining). Η φράση «Εισαγωγή Υπηρεσιών» αναφέρεται στην

δυνατότητα οι ροές κίνησης να οδηγούνται δυναμικά σε ένα φυσικό ή εικονικό server όπου παρέχει μια από τις επιπέδου 4-7 υπηρεσίες. Η φράση «Αλυσιδωτή σύνδεση υπηρεσιών» αναφέρεται στη δυνατότητα οι ροές κίνησης να οδηγούνται δυναμικά μέσα από μια σειρά φυσικών ή εικονικών servers που παρέχουν τις ίδιες υπηρεσίες.

Υπηρεσίες Ασφάλειας

Εξαιτίας της δυνατότητας ταιριάσματος ροών επιπέδου 2-4, οι μεταγωγείς OpenFlow μπορούν να κάνουν φιλτράρισμα των πακέτων όπως μπαίνουν στο δίκτυο, δρώντας έτσι σαν ένα απλό τείχος προστασίας στην άκρη του δικτύου. Με OpenFlow μεταγωγείς που υποστηρίζουν τροποποίηση των κεφαλίδων των πακέτων ένας ελεγκτής OpenFlow θα μπορεί επίσης να κάνει το μεταγωγέα να ανακατευθύνει συγκεκριμένες ύποπτες ροές σε ελέγχους ασφαλείας υψηλότερου επιπέδου όπως IDS/IPS συστήματα, εφαρμογές τείχους προστασίας, και συσκευές DPS (Data Loss Prevention). Άλλες εφαρμογές ασφάλειας του OpenFlow ελεγκτή μπορούν να συσχετίσουν ύποπτες ροές σε βάσεις δεδομένων με χαρακτηριστικά κακόβουλων λογισμικών (databases of malware signatures) ή να αποτρέψουν DDoS επιθέσεις.

Υπηρεσίες Εξισορρόπισης Φόρτου (Load Balance Services)

Η δυνατότητα τροποποίησης της κεφαλίδας ενός πακέτου του OpenFlow θα επιτρέψει επίσης τους μεταγωγείς να λειτουργούν σαν μια απλή και συμφέρουσα συσκευή που παράλληλα θα εξυπηρετεί και σαν ισορροπιστής φόρτου (load balancer). Με την λειτουργία τροποποίησης μια νέα ροή μπορεί να αποτελεί μια νέα είσοδο στον πίνακα ροών ο οποίος θα περιέχει μια ενέργεια τροποποίησης της MAC και IP διεύθυνσης προορισμού. Η τροποποιημένη διεύθυνση μπορεί να χρησιμοποιηθεί για να κατευθύνει κίνηση στον server που θα έχει επιλεγεί από την εφαρμογή ισορρόπησης φόρτου του ελεγκτή.

Το Πανεπιστήμιο της Ιντιάνα (IU) έχει αναπτύξει μια νέα εφαρμογή ισορρόπησης φόρτου η οποία είναι βασισμένη στο OpenFlow, την FlowScale. Σύμφωνα με το πανεπιστήμιο «Η FlowScale παρέχει πολύπλοκη, διαμοιρασμένη ισορροπία φόρτου δικτυακής κίνησης χρησιμοποιώντας ένα OpenFlow-capable Top of Rach (ToR) switch.» Το πανεπιστήμιο ανέπτυξε την εφαρμογή στο σύστημα εντοπισμού εισβολών του (IDS) για να διαμοιράσει κίνηση ίσα σε αισθητήρες. Η εφαρμογή για την ώρα αναπτύσσεται σαν μέρος του IDS που χειρίζεται το Γραφείο Ασφάλειας του Πανεπιστημίου της Ιντιάνα.

5.3 Θέματα στην εφαρμογή του SDN

Λόγω του ότι το SDN είναι ακόμα μια νέα τεχνολογία υπάρχουν αρκετά θέματα με λίγες καθοριστικές λύσεις. Στις παρακάτω παραγράφους περιγράφονται κάποιοι από του κυριότερους προβληματισμούς σε ζητήματα διαλειτουργικότητας, επεκτασιμότητας, απόδοσης και ασφάλειας.

Διαλειτουργικότητα
Ενώ το OpenFlow έχει τυποποιηθεί ως το νότιο API, δεν υπάρχει ακόμα κάποιο πρότυπο για το βόρειο API επιτρέποντας έτσι σε κάθε ελεγκτή να καθορίσει τη δική του μοναδική διεπαφή. Χωρίς προτυποποίηση οι προγραμματιστές ίσως χρειαστούν να προσαρμόζουν τις εφαρμογές σε διαφορετικούς ελεγκτές. Παρόλα αυτά η ONF εστιάζει στην προτυποποίηση μερικών βόρειων διεπαφών μέσα από το North Bound Interface Working Group (NBI-WG). Ενώ το NB-WG δεν έχει δώσει ακόμα τις κατευθυντήριες γραμμές για την προτυποποίηση του βόρειου API, ο σχηματισμός αυτής της ομάδας εργασίας αποδεικνύει την αφοσίωση της κοινότητας SDN για την επίλυση αυτού του ζητήματος.

Ένα άλλο θέμα διαλειτουργικότητας έχει να κάνει με τον εξοπλισμό. Για τα νέα δίκτυα η εφαρμογή του SDN είναι αρκετά απλή μιας και όλες οι συσκευές υποστηρίζουν τη νέα τεχνολογία. Η μετάβαση από ένα παλαιότερο δίκτυο μπορεί να είναι δύσκολη. Για το λόγο αυτό ένας οργανισμός που θέλει να μεταβεί από τον παραδοσιακό τρόπο δικτύωσης στο SDN απαιτεί μια περίοδο διαλειτουργικότητας με υβριδικό εξοπλισμό. Δικτυακοί κόμβοι τόσο παραδοσιακοί αλλά και τεχνολογίας SDN μπορούν να λειτουργούν μαζί με τη βοήθεια ενός κατάλληλου πρωτοκόλλου που υποστηρίζει επικοινωνίες SDN ενώ παράλληλα παρέχει backward compatibility με υπάρχουσες IP και MPLS τεχνολογίες, κάτι που μειώνει κόστος, κίνδυνο και διαταραχή υπηρεσιών όσο γίνεται η μετάβαση στο SDN.

Επεκτασιμότητα
Δεδομένου ότι η αρχιτεκτονική SDN περιλαμβάνει συγκεντρωμένους ή εν μέρει διαμοιρασμένους ελεγκτές που συνδέονται με το επίπεδο δεδομένων πολλαπλών συσκευών, υπάρχει η πιθανότητα οι ελεγκτές να μετατραπούν σε σημεία συμφόρησης. Συγκεκριμένα, δίκτυα με μεγάλους όγκους δικτυακών αιτημάτων μπορούν να κυριεύσουν τους ελεγκτές. Καθώς τα δίκτυα μεγαλώνουν, το ίδιο ισχύει και για τη συμφόρηση και έτσι οι επιδόσεις του δικτύου υποβαθμίζονται.

Απόδοση
Η απόδοση είναι το μεγαλύτερο ζήτημα για όλα τα δίκτυα. Ανεξάρτητα από το πόσο ισχυρό, ασφαλές, επεκτάσιμο, ή διαλειτουργικό είναι ένα δίκτυο είναι άχρηστο αν δεν έχει απόδοση.

Σε μια αρχιτεκτονική με ξεχωριστό επίπεδο ελέγχου και δεδομένων μπορεί να δημιουργηθεί καθυστέρηση (latency). Σε μεγάλα δίκτυα αυτό μπορεί να οδηγήσει σε ένα απαράδεκτο επίπεδο καθυστέρησης υποβαθμίζοντας έτσι την απόδοση του δικτύου. Επίσης, ο χρόνος απόκρισης του ελεγκτή και η ρυθμοαπόδοση μπορούν να συμβάλουν στην κακή απόδοση καθώς και να προκαλέσουν θέματα επεκτασιμότητας.

Η λύση σε πολλά ζητήματα επιδόσεων σε μεγάλα και συνεχώς αυξανόμενα δίκτυα είναι να δοθεί περισσότερη νοημοσύνη στο επίπεδο των δεδομένων ή να υιοθετηθεί μια αρχιτεκτονική κατανεμημένου επιπέδου ελέγχου. Ενώ αυτό μπορεί να βελτιώσει την απόδοση του SDN, κινείται κάπως μακριά από την πρόθεση του και αναπαράγει τα παραδοσιακά δίκτυα που είναι χτισμένα σε πλήρως κατανεμημένης νοημοσύνης συσκευές. Πρέπει να βρεθεί μια ισορροπημένη λύση όπου η εικονοποίηση να διατηρείται χωρίς να υποβαθμίζει την απόδοση του δικτύου.

Ασφάλεια

Μιας και το επίπεδο ελέγχου είναι μια τόσο κεντρική λειτουργία σε μια αρχιτεκτονική SDN, οι στρατηγικές ασφάλειας πρέπει να επικεντρωθούν στην προστασία του ελεγκτή και στην πιστοποίηση της αυθεντικότητας της πρόσβασης μιας εφαρμογής στο επίπεδο ελέγχου.

Νέες υπηρεσίες μπορούν να αποτελέσουν απειλές για την ασφάλεια ενός δικτύου καθώς προγραμματιστές και διαχειριστές μπορούν να εισαγάγουν ασυναίσθητα κακό κώδικα (at-risk code) κάτι που λόγω του κεντρικού ελεγκτή θα έχει ως συνέπεια την επέκταση της απειλής σε όλο το δίκτυο. Επίσης, η εικονική φύση του SDN μπορεί να οδηγήσει στη δημιουργία αμέτρητων τμημάτων του δικτύου, το καθένα με τις δικές του απαιτήσεις κινδύνων και ασφάλειας. Μια πιο εκτεταμένη επισκόπηση των επιπτώσεων αλλά και των πλεονεκτημάτων που επιφέρει η αρχιτεκτονική SDN στην ασφάλεια των δικτύων περιγράφεται στην επόμενη ενότητα.

6. Ασφάλεια και SDN

Η αρχιτεκτονική SDN χωρίζει το επίπεδο δεδομένων από το επίπεδο του ελέγχου σε δρομολογητές και μεταγωγείς. Το επίπεδο δεδομένων, το οποίο ήταν πάντοτε ιδιωτικό και γνωστό μόνο στους προμηθευτές, πλέον ελέγχεται κεντρικά ενώ στέλνει εντολές και νοημοσύνη στο επίπεδο δεδομένων. Η ικανότητα αυτή, της διαχείρισης του επιπέδου ελέγχου μέσω ανοικτών πρωτοκόλλων όπως το OpenFlow, επιτρέπει ακριβείς αλλαγές στις συσκευές του δικτύου, κάτι που αυξάνει την ταχύτητα και την ασφάλεια. Όπως πάντα, η εισαγωγή μιας νέας τεχνολογίας φέρνει πλεονεκτήματα αλλά και προβληματισμούς.

6.1 Πλεονεκτήματα ασφαλείας στο SDN

Μέσα από το SDN οι διαχειριστές μπορούν να αλλάξουν τους κανόνες, και έχοντας μια γρήγορη, υψηλού επιπέδου ματιά σε όλους τους τομείς του δικτύου είναι σε θέση να το τροποποιήσουν. Αυτή η ελευθερία και ο έλεγχος επιτρέπει επίσης την καλύτερη ασφάλεια των συστημάτων. Έχοντας τη δυνατότητα να ελέγξουν και να δουν το εσωτερικό του δικτύου από μια κεντρική άποψη, οι διαχειριστές μπορούν να κάνουν αλλαγές αποδοτικά. Για παράδειγμα, αν υπήρχε ένα malware εντός του δικτύου, με το SDN και το OpenFlow ο διαχειριστής θα ήταν σε θέση να περιορίσει γρήγορα το ξέσπασμα, από ένα συγκεντρωτικό επίπεδο ελέγχου που θα σταματούσε την κίνηση, χωρίς να χρειάζεται να έχει πρόσβαση σε πολλαπλούς δρομολογητές ή μεταγωγείς. Η ικανότητα να αλλάζει γρήγορα διάφορους παράγοντες στο δίκτυο επιτρέπει στους διαχειριστές να εκτελούν διαμόρφωση κίνησης (traffic shaping) και QoS των πακέτων με πιο ασφαλή τρόπο. Αυτή η δυνατότητα υπάρχει και τώρα, αλλά δεν υπάρχει η ταχύτητα και η αποτελεσματικότητα και αυτό θα περιορίσει την ικανότητα του διαχειριστή όταν προσπαθεί να ασφαλίσει το δίκτυο. Το SDN προσφέρει μια σειρά από χαρακτηριστικά που είναι κατάλληλα για την εφαρμογή ενός εξαιρετικά ασφαλούς και διαχειρίσιμου περιβάλλοντος:

- Το μεγαλύτερο όφελος της ασφαλείας βασισμένης στο SDN είναι το επιλεκτικό μπλοκάρισμα μόνο της κακόβουλης κίνησης ενώ εξακολουθεί να επιτρέπεται η κανονική ροή της υπόλοιπης κίνησης. Επιπλέον, εφαρμογές ασφάλειας SDN είναι σε θέση να ενεργούν σε οποιεσδήποτε ανωμαλίες με την εκτροπή συγκεκριμένων ροών σε υπηρεσίες ασφαλείας, όπως τείχη προστασίας ή συστήματα ανίχνευσης/πρόληψης εισβολής (IPS/IDS). Το SDN έχει τη δυνατότητα επίτευξης μεγαλύτερης προβολής της ασφάλειας των δικτύων και επιτάχυνσης του ρυθμού της εφαρμογής νέων υπηρεσιών ασφαλείας.

- Το πρότυπο ροών (flow paradigm) είναι ιδανικό για την επεξεργασία της ασφάλειας, διότι προσφέρει ένα μοντέλο σύνδεσης που εκτείνεται στο δίκτυο από άκρο σε άκρο και που δεν δεσμεύεται από τους περιορισμούς της παραδοσιακής δρομολόγησης.

- Σε SDN αρχιτεκτονικές η νοημοσύνη του δικτύου είναι κεντρική, έτσι που η λήψη αποφάσεων διευκολύνεται βασιζόμενη σε μια ολική ή μερική όψη του δικτύου, σε αντίθεση με τα σημερινά δίκτυα όπου οι κόμβοι δεν γνωρίζουν τη συνολική κατάσταση του δικτύου. Αυτό δίνει στους χειριστές ένα πιο αποτελεσματικό τρόπο για τον εντοπισμό και την απομόνωση των απειλών.

- Μια πολιτική ασφαλείας βασισμένη στους πόρους του δικτύου επιτρέπει ενοποιημένη διαχείριση των διαφόρων συσκευών, από εξαιρετικά ασφαλή τείχη προστασίας και συσκευές ασφαλείας μέχρι συσκευές πρόσβασης.

- Με την ανάμειξη παλαιότερων και πραγματικού χρόνου δεδομένων κατάστασης και απόδοσης του δικτύου το SDN διευκολύνει την έξυπνη λήψη αποφάσεων, την ευελιξία, την απλότητα λειτουργίας και βελτιώνει την ασφάλεια. Με τη σωστή διαμόρφωση μπορεί να εξασφαλιστεί ένα περιβάλλον SDN χωρίς αντίκτυπο στις δυνατότητες του. Αυτό παρέχει ένα τρόπο για να αντιμετωπισθεί μια από τις πιο κοινές κριτικές για την ασφάλεια, ότι δηλαδή η ασφάλεια είναι πρόσθετη και όχι ενσωματωμένη.

6.2 Ανησυχίες για την ασφάλεια SDN (Security Concerns)

Στην ενότητα αυτή περιγράφονται μερικά θέματα που σχετίζονται με την ασφάλεια κατά την εφαρμογή της αρχιτεκτονικής SDN. Η πλειοψηφία των ανησυχιών για την ασφάλεια δικτύων προσδιοριζόμενων από λογισμικό πρόκειται να εξελιχθεί γύρω από τον ίδιο τον ελεγκτή. Ο ελεγκτής θεωρείται ο εγκέφαλος της διαδικασίας δρομολόγησης. Αυτό σημαίνει πως η μεγαλύτερη πρόκληση για τους διαχειριστές της ασφάλειας στα δίκτυα SDN είναι η ασφάλιση του ελεγκτή με κάθε κόστος. Τώρα που η νοημοσύνη έχει φύγει από τους δρομολογητές και τους μεταγωγείς και έχει μεταφερθεί στον ελεγκτή, είναι σημαντικό να πληρούνται οι παρακάτω προϋποθέσεις:

Η γνώση και ο έλεγχος του ποιος έχει πρόσβαση στον ελεγκτή και που βρίσκεται στο δίκτυο είναι μια μεγάλη ανησυχία όσον αφορά στην ασφάλεια. Είναι επίσης σημαντικό να τονιστεί ότι η πρόσβαση στον ελεγκτή θα μπορούσε ενδεχομένως να δώσει τον πλήρη έλεγχο σε έναν εισβολέα, έτσι είναι ζωτικής σημασίας να είναι ασφαλισμένη. Κρίνεται απαραίτητο να επιβεβαιώνεται η ασφάλεια μεταξύ του ελεγκτή και του τελικού κόμβου (δρομολογητής ή μεταγωγέας) συγκεκριμένα να επιβεβαιώνεται πως επικοινωνούν μέσω SSL για να αποφευχθεί οποιαδήποτε κακόβουλη πρόθεση πρόσβασης στον ελεγκτή. Τέλος, ο διαχειριστής πρέπει να βεβαιωθεί ότι οτιδήποτε βγαίνει από το σύστημα καταγράφεται. Κατά την εφαρμογή του SDN πρέπει να ενημερωθεί το σύστημα πρόληψης εισβολών (IPS) του οργανισμού καθώς και όποια άλλη τεχνολογία φιλτραρίσματος που μπορεί να μπλοκάρει ή να καταγράφει αλλαγές.

Εν κατακλείδι, το SDN είναι μια αναδυόμενη τεχνολογία που εφαρμόζει την ασφάλεια δίνοντας σε ένα διαχειριστή μια πλήρη εικόνα του δικτύου μέσω του ελεγκτή. Ο ελεγκτής αυτός είναι ο εγκέφαλος των SDN, και χωρίς την κατάλληλη ασφάλεια γύρω του, το δίκτυο γίνεται εντελώς ευάλωτο σε κακόβουλες επιθέσεις ή τυχαίες αλλαγές, οι οποίες θα μπορούσαν και να το «ρίξουν». Για το λόγο αυτό είναι πολύ σημαντικό να διασφαλιστεί ότι η ασφάλεια είναι πρωταρχικό μέλημα στο σχεδιασμό, την υλοποίηση και τη διαχείριση των SDNs.

6.3 Στόχοι επίθεσης στα δίκτυα SDN

Ο κεντρικός ελεγκτής και η νότια διεπαφή είναι εξαιρετικοί στόχοι για έναν εισβολέα, ο οποίος μπορεί εύκολα να τροποποιήσει κώδικα SDN και να κάνει αλλαγές στον έλεγχο της SDN κυκλοφορίας και έτσι να θέσει σε κίνδυνο την ασφάλεια του δικτύου. Στις παρακάτω παραγράφους αναλύεται γιατί τα σημεία αυτά είναι ευάλωτα σε απειλές που θα μπορούσαν να υποβαθμίσουν τη διαθεσιμότητα, την απόδοση και την ακεραιότητα του δικτύου.

SDN ελεγκτής ως στόχος: Ο ελεγκτής SDN είναι ο πρωταρχικός στόχος για τους εισβολείς, μιας και είναι τόσο το σημείο της επιρροής σε ένα δίκτυο όσο και ένα κεντρικό σημείο αποτυχίας. Υπάρχουν τόσες πολλές ευκαιρίες για έναν εισβολέα να κάνει αλλαγές σε ολόκληρη την υποστήριξη της συμπεριφοράς της κίνησης του δικτύου SDN απλά τροποποιώντας τον ελεγκτή. Αυτό δεν είχε υπάρξει ποτέ δυνατό γιατί τα παραδοσιακά εργαλεία διαχείρισης δικτύου δεν έδιναν τόσο μεγάλη ευελιξία στην δυναμική αλλαγή της συμπεριφοράς του δικτύου. Η δυνατότητα προγραμματισμού των ελεγκτών SDN είναι ένα δίκοπο μαχαίρι. Οι διαχειριστές μπορούν να εγκαταστήσουν εφαρμογές ασφαλείας στη βόρεια διεπαφή του ελεγκτή για να δημιουργήσουν νέους τρόπους εφαρμογής πολιτικών ασφάλειας. Οι εφαρμογές αυτές συμβουλεύουν τον ελεγκτή να χρησιμοποιήσει τους μεταγωγείς και δρομολογητές που ελέγχει ως σημεία επιβολής της πολιτικής. Ωστόσο, αυτή η προγραμματιζόμενη βόρεια διεπαφή είναι επίσης ευάλωτη γιατί αυτές οι εφαρμογές είναι οι ίδιες που μπορούν να επαναπρογραμματίσουν το δίκτυο μέσω του ελεγκτή. Οι εισβολείς μπορούν να ξεγελάσουν τους διαχειριστές ώστε να εγκαταστήσουν εφαρμογές που θέτουν την ασφάλεια σε κίνδυνο. Με επαρκείς γνώσεις σχετικά με τις καλοήθεις εφαρμογές που εκτελούνται σε έναν ελεγκτή, ένας εισβολέας θα μπορούσε να κάνει το δίκτυο να κάνει κάτι εντελώς απροσδόκητο απλά με την αποστολή μιας προσεκτικά δημιουργημένης ροής πακέτων.

Οι εφαρμογές OpenFlow μπορεί να αντιφάσκουν μεταξύ τους. Μπορούν να εισάγουν κανόνες που, όταν συνδυάζονται, έχουν μερικές ενδιαφέρουσες αλληλεπιδράσεις που κάποιος δεν θα περίμενε ή θα ήθελε να συμβούν. Γενικά, οι ελεγκτές SDN στερούνται της πολυτέλειας να κατανοήσουν ότι οι εφαρμογές ασφαλείας θα πρέπει να έχουν προτεραιότητα σε σχέση με άλλες εφαρμογές που επικοινωνούν. Ακόμη και μια ακίνδυνη

εφαρμογή μπορεί να σπάσει τις πολιτικές ασφαλείας, εάν ο ελεγκτής δεν καταλαβαίνει πώς να χειριστεί τις αιτήσεις εφαρμογών που έρχονται σε αντίθεση με τις πολιτικές ασφαλείας. Για παράδειγμα, ας θεωρηθεί πως μια εφαρμογή ασφαλείας OpenFlow αποφασίζει πως μια εσωτερική μηχανή λειτουργεί με έναν τρόπο που την κάνει να πιστεύει ότι έχει μολυνθεί. Η εφαρμογή ασφαλείας βάζει σε καραντίνα αυτή τη μηχανή και καταργεί την ικανότητά της να επικοινωνεί με το δίκτυο. Την ίδια στιγμή, μια εφαρμογή εξισορρόπησης φόρτου μπορεί να δει αυτή τη μηχανή και να πει ότι είναι η λιγότερο φορτωμένη στο δίκτυο. Έτσι, η εφαρμογή εξισορρόπησης φόρτου αποφασίζει να ξεκινήσει την εκτροπή της κυκλοφορίας σε αυτή τη μηχανή, την ίδια που είχε τεθεί σε καραντίνα.

Ασφάλεια νότιας διεπαφής: Η ONF προσδιόρισε την επικοινωνία της νότιας διεπαφής με τους ελεγκτές και τις συσκευές δεδομένων προώθησης ως ευάλωτη. Πρωτόκολλα νότιας διεπαφής όπως το OpenFlow έχουν τεχνολογία ελέγχου ταυτότητας που εμποδίζει έναν εισβολέα να πλαστογραφήσει εντολές ροής από έναν ελεγκτή σε ένα μεταγωγέα. Επίσης, οι προμηθευτές του SDN έχουν εφαρμόσει πιστοποιητικά ταυτότητας και μεταξύ των ελεγκτών και των μεταγωγέων. Βέβαια, ένας εισβολέας κατά πάσα πιθανότητα δεν θα προσπαθήσει να καταλάβει τη νότια διεπαφή γιατί υπάρχουν ευκολότεροι στόχοι. θα μπορούσε να στοχεύσει ελεγκτές, μεταγωγείς ή ακόμα και εικονικούς μεταγωγείς με επιθέσεις denial-of-service. Υπάρχουν τρόποι που μπορεί κανείς να γεμίσει και να διατηρήσει το επίπεδο ελέγχου απασχολημένο ή να καταστήσει τη διεπαφή μεταξύ του επιπέδου ελέγχου και δεδομένων κορεσμένη και να επιβραδύνει ολόκληρο το δίκτυο. Έχουν παρατηρηθεί μοντέλα όπου ο εισβολέας δεν προσπαθεί να ρίξει το δίκτυο, αλλά δημιουργεί σειρές πακέτων για να κορέσει τις αλληλεπιδράσεις μεταξύ του επιπέδου δεδομένων και ελέγχου.

Άλλοι πιθανοί στόχοι είναι:

- Οι τελικοί χρήστες: οι τελικοί χρήστες είναι από τους πιο επιθυμητούς στόχους. Αυτό ισχύει κυρίως επειδή αυτοί είναι που έχουν την πληροφορία, υπηρεσία ή λειτουργία για κλοπή, παραποίηση ή δολιοφθορά. Εάν ένας εισβολέας αποκτήσει πρόσβαση στο στόχο, θα προσπαθήσει να βρει τυχόν τρωτά σημεία για να επιτεθεί.
- Το δίκτυο: Οι παράγοντες επιβολής ασφαλείας που παρουσιάζονται σε κάθε ένα από τα πέντε σημεία εστίασης της ασφάλειας (βλ. Πίνακα 1), μπορούν να περιέχουν τρωτά σημεία και σφάλματα κατά την εφαρμογή τους. Ο στόχος αυτός παρουσιάζει την ικανότητα να αλλάξει τις ρυθμίσεις του δικτύου και για παράδειγμα να αποτρέψει τους τελικούς χρήστες από τη χρήση μιας υπηρεσίας. Ένας εισβολέας θα προσπαθήσει να εκθέσει οποιαδήποτε ευπάθεια για να πάρει τον έλεγχο του δικτύου ή κάποιο μέρος του.

- Η κίνηση: Καθώς οι hosts επικοινωνούν, κρίσιμες πληροφορίες ταξιδεύουν μέσω του δικτύου. Ο στόχος αυτός δίνει την δυνατότητα σε κάποιον να «κρυφακούει» την κυκλοφορία και να κλέψει το περιεχόμενό της.

Επίπεδο Εφαρμογών	- Επιβεβαίωση της προέλευσης της εφαρμογής - Επιβεβαίωση της αξιοπιστίας του ελεγκτή - Απαίτηση άδειας πρόσβασης για την τροποποίηση και την παρακολούθηση της εφαρμογής - Η επικοινωνία μεταξύ εφαρμογών πρέπει να γίνεται μόνο μέσω του ελεγκτή ώστε να υποστηρίζεται η αξιοπιστία της εφαρμογής
Βόρια Διεπαφή	- Περιορισμός πρόσβασης στη διεπαφή μόνο σε εγκεκριμένες εφαρμογές - Η επικοινωνία μεταξύ εφαρμογών πρέπει να γίνεται μέσω του ελεγκτή
Επίπεδο Ελέγχου	- Παροχή πληροφοριών δικτύου μόνο σε ταυτοποιημένες εφαρμογές - Δημιουργία διαφορετικών επιπέδων πληροφοριών δικτύου - Παροχή προνομίων σε εφαρμογές σύμφωνα με τα προαναφερθέντα επίπεδα - Χρήση ασφαλούς επικοινωνίας OpenFlow (TLS) μεταξύ μεταγωγέων - Έλεγχος αιτημάτων κανόνων ροής και αποτροπή συγκρούσεων ή παραβιάσεων ασφαλείας - Παροχή παραπάνω ελεγκτή σε περίπτωση αποτυχίας - Απαίτηση άδειας πρόσβασης για την τροποποίηση ή παρακολούθηση του ελεγκτή
Νότια Διεπαφή	- Χρήση ασφαλούς επικοινωνίας του OpenFlow (κρυπτογράφηση και έλεγχος ταυτότητας)
Επίπεδο Δεδομένων	- Πιστοποίηση από και προς τον ελεγκτή - Απαίτηση άδειας πρόσβασης για την τροποποίηση ή παρακολούθηση του μεταγωγέα - Ασφάλιση καλωδίωσης του δικτύου με περιορισμένη πρόσβαση

Πίνακας 1: Πιθανοί παράγοντες ενίσχυσης ασφάλειας

Σε γενικές γραμμές, στον τομέα της τεχνολογίας των πληροφοριών, οι ευπάθειες λογισμικού ανακαλύπτονται είτε κατά τη διάρκεια της ανάπτυξης είτε μέσω της έκθεσης από μια εξωτερική οντότητα (ένας εισβολέας ή ένας χρήστης). Ως εκ τούτου, ένα

σημαντικό ποσό των πόρων χρησιμοποιείται για δοκιμή κατά την ανάπτυξη λογισμικού. Οι στόχοι αυτοί παρέχουν τα μέσα για το σχεδιασμό περιπτώσεων δοκιμών για τον έλεγχο

ασφαλείας, κατά την ανάπτυξη λογισμικού SDN και ιδιαίτερα κατά το σχεδιασμό του ελεγκτή SDN.

6.4 Υπάρχουσες Λύσεις Ασφαλείας

Σχεδόν το σύνολο των λύσεων ασφάλειας δικτύων σήμερα περιλαμβάνει:

1. Τείχη προστασίας για περιμετρική άμυνα και έλεγχο του εσωτερικού τομέα.
2. Συστήματα ανίχνευσης και πρόληψης εισβολής που παρακολουθούν τις δραστηριότητες του δικτύου για κακόβουλες δραστηριότητες ή παραβιάσεις της πολιτικής και προσπαθούν να αποτρέψουν τις επιθέσεις.
3. Εικονικά ιδιωτικά δίκτυα Secure Sockets Layer (SSL VPN).
4. Λύσεις διαχείρισης δικτύου (network management solutions) που επιχειρούν να διαχειριστούν κεντρικά πολλές λειτουργίες ασφάλειας μέσω κονσόλας.
5. IEEE 802.1X port-based αυθεντικοποίηση δικτύου και ελέγχου πρόσβασης.
6. Ασφάλεια IP για end-to-end έλεγχο ταυτότητας και κρυπτογράφηση των πακέτων IP .
7. Πρωτόκολλο Transport Layer Security (TLS).
8. Remote Access Dial In User Service (RADIUS) πρωτόκολλο δικτύωσης, το οποίο προσφέρει κεντρική διαχείριση πιστοποίησης, εξουσιοδότησης και λογιστικής (AAA) για τις τελικές συσκευές.

6.5 Βήματα για ασφάλεια στο SDN

Η ασφάλεια πρέπει να είναι παντού μέσα σε ένα δίκτυο SDN. Πρέπει να υπάρχει ενσωματωμένη στην αρχιτεκτονική, καθώς και να παραδίδεται ως υπηρεσία για την προστασία της διαθεσιμότητας, της ακεραιότητας και της ιδιωτικότητας όλων των συνδεδεμένων πόρων και των πληροφοριών. Το SDN δίνει τη δυνατότητα να προγραμματιστεί εύκολα το δίκτυο. Ωστόσο είναι αυτό το ίδιο το πλεονέκτημα που μπορεί να οδηγήσει σε ευπάθειες ασφαλείας. Μέσα σε αυτό το δυναμικό περιβάλλον, είναι ζωτικής σημασίας η πολιτική ασφάλειας του δικτύου να ενισχύεται. Το SDN διευκολύνει την ταχεία καινοτομία στο επίπεδο ελέγχου του δικτύου, παρέχοντας μια προγραμματιζόμενη υποδομή δικτύου για τον υπολογισμό πολιτικών ροών, αν και όταν ζητηθεί (on-demand). Ωστόσο, ο δυναμισμός των προγραμματιζόμενων δικτύων εισάγει επίσης νέες προκλήσεις ασφαλείας που απαιτούν καινοτόμες λύσεις. Μια κρίσιμη πρόκληση είναι η αποτελεσματική ανίχνευση και η συμφωνία των ίσως αντικρουόμενων κανόνων ροής που επιβάλλονται από δυναμικές εφαρμογές OpenFlow. Καθώς το SDN διαχωρίζει το επίπεδο δεδομένων από το επίπεδο ελέγχου αυτό σημαίνει ότι όλη η νοημοσύνη φεύγει από τους δρομολογητές ή μεταγωγείς και είναι κεντρική. Ο κεντρικός έλεγχος που παρέχεται από το SDN θα οδηγήσει τελικά σε δρομολόγηση

προσδιορισμένη από αστάθεια (insecurity-defined routing) και άλλες στρατηγικές ασφαλείας που θα μπορούσαν να αλλάξουν για πάντα το πώς κάποιος θα υπερασπίζεται το δίκτυο και τις εφαρμογές ή τα δεδομένα που τρέχουν πάνω σε αυτό. Για την ασφάλιση του SDN πρέπει:

1. Να ασφαλιστεί ο ελεγκτής: Ως κεντρικό σημείο αποφάσεων, η πρόσβαση στον ελεγκτή πρέπει να ελέγχεται αυστηρά.
2. Να προστατευθεί ο ελεγκτής: Εάν ο ελεγκτής «πέσει» (για παράδειγμα, εξαιτίας μιας επίθεσης DDoS), «πέφτει» και το δίκτυο, κάτι που σημαίνει ότι η διαθεσιμότητα του ελεγκτή πρέπει να διατηρηθεί.
3. Να εδραιωθεί η εμπιστοσύνη: Η προστασία των επικοινωνιών σε όλο το δίκτυο είναι κρίσιμη. Αυτό απαιτεί τη διασφάλιση πως ο ελεγκτής, οι εφαρμογές που φορτώνονται σε αυτόν, καθώς και οι συσκευές που διαχειρίζεται είναι όλες οι αξιόπιστες οντότητες που λειτουργούν όπως θα έπρεπε.
4. Να δημιουργηθεί ένα σταθερό πλαίσιο πολιτικής: Χρειάζεται ένα σύστημα ελέγχου και εξισορρόπησης για να βεβαιώνει ότι ο ελεγκτής κάνει αυτό που πραγματικά πρέπει να κάνει.
5. Να γίνεται διεξαγωγή έρευνας και αποκατάστασης: Όταν κάτι πάει στραβά, το σύστημα πρέπει να μπορεί να προσδιορίσει τι ήταν αυτό, να γίνει αποκατάσταση, ενδεχομένως να σταλθεί αναφορά για αυτό και στο τέλος να μπορεί το σύστημα να προστατευθεί από αυτό στο μέλλον.

Πέρα από την ίδια την αρχιτεκτονική το πώς πρέπει η ασφάλεια να αναπτυχθεί, να διαχειριστεί, και να ελέγχεται σε ένα περιβάλλον SDN δεν έχει ακόμα προσδιοριστεί. Υπάρχουν διάφορες προσεγγίσεις που ορισμένοι πιστεύουν ότι η ασφάλεια είναι καλύτερο να υπάρχει ενσωματωμένη εντός του δικτύου, ενώ άλλοι υποστηρίζουν ότι είναι καλύτερο να ενσωματώνεται σε διακομιστές, συστήματα αποθήκευσης και άλλες συσκευές. Ανεξάρτητα από αυτό, οι λύσεις που πρέπει να σχεδιαστούν οφείλουν να δημιουργήσουν ένα περιβάλλον πιο επεκτάσιμο, αποτελεσματικό και ασφαλές. Πρέπει να είναι:

- Απλές για την ανάπτυξη, διαχείριση και συντήρηση στο άκρως δυναμικό περιβάλλον SDN,
- Οικονομικά αποδοτικές για την διασφάλιση πως η ασφάλεια μπορεί να αναπτυχθεί παντού και,
- Ασφαλείς για την προστασία από τις πιο πρόσφατα προηγμένες, στοχευμένες απειλές που αντιμετωπίζει το κάθε σύστημα

6.6 Ασφάλεια στα δίκτυα SDN

Λαμβάνοντας υπόψη όλες τις παραπάνω προκλήσεις και τις δυσκολίες μια νέα κατηγορία εμφανίζεται για την ασφάλεια, που ονομάζεται «Ασφάλεια προσδιοριζόμενη από Λογισμικό» (Software Defined Security, SDSec), η οποία προσφέρει την εφαρμογή της ασφάλειας στο δίκτυο, διαχωρίζοντας τον έλεγχο ασφαλείας από την επεξεργασία της

ασφάλειας και τα επίπεδα προώθησης, παρόμοια με τον τρόπο που το SDN αφαιρεί το επίπεδο ελέγχου του δικτύου από το επίπεδο προώθησης. Το αποτέλεσμα είναι ένα δυναμικά κατανεμημένο σύστημα που εικονοποιεί τη λειτουργία της εφαρμογής της ασφάλειας του δικτύου.Το SDSec είναι ένα παράδειγμα εικονοποίησης των λειτουργιών του δικτύου (NFV), το οποίο προσφέρει ένα νέο τρόπο για το σχεδιασμό, την ανάπτυξη και τη διαχείριση των υπηρεσιών δικτύωσης. Αυτό το επιτυγχάνει με την αποσύνδεση της λειτουργίας του δικτύου (όπως τα firewalls και την ανίχνευση εισβολών) από τις ιδιόκτητες συσκευές υλικού (proprietary hardware), έτσι ώστε να μπορεί να τρέξει στο λογισμικό. Έχει σχεδιαστεί για να εδραιώσει και να παραδώσει τα στοιχεία δικτύωσης που απαιτούνται για να υποστηρίξουν πλήρως εικονοποιημένες υποδομές, συμπεριλαμβανομένων των εικονικών servers, των συστημάτων αποθήκευσης, ακόμη και άλλων δικτύων. Έτσι, από την παραπάνω σύντομη εισαγωγή, είναι σαφές ότι το SDSec που παρέχει την ασφάλεια δικτύου στα SDNs μπορεί να ταξινομηθεί περαιτέρω σε δύο μέρη:

- Η ασφάλεια ως μια εφαρμογή (SaaA) ή Εφαρμογές Ασφάλειας με SDN και
- Ασφάλεια ως μια υπηρεσία (SaaS) ή Εφαρμογές Ασφάλειας στο SDN.

6.6.1 Εφαρμογές ασφαλείας με SDN

Το FRESCO (Framework for Enabling Security Controls in OpenFlow) είναι ένα framework ανάπτυξης εφαρμογών ασφάλειας του OpenFlow. Διευκολύνει την ταχεία σχεδίαση και σύνθεση εφαρμογών βασιζόμενων στο OpenFlow για την ανίχνευση και τον μετριασμό του κινδύνου, δημιουργώντας έτσι ένα πιο γρήγορο και συνεργατικό περιβάλλον για προγραμματιστές που επικεντρώνονται στην ασφάλεια. Το FRESCO λειτουργεί με τον ελεγκτή NOX, αλλά μπορεί να επεκταθεί και σε άλλους ελεγκτές.

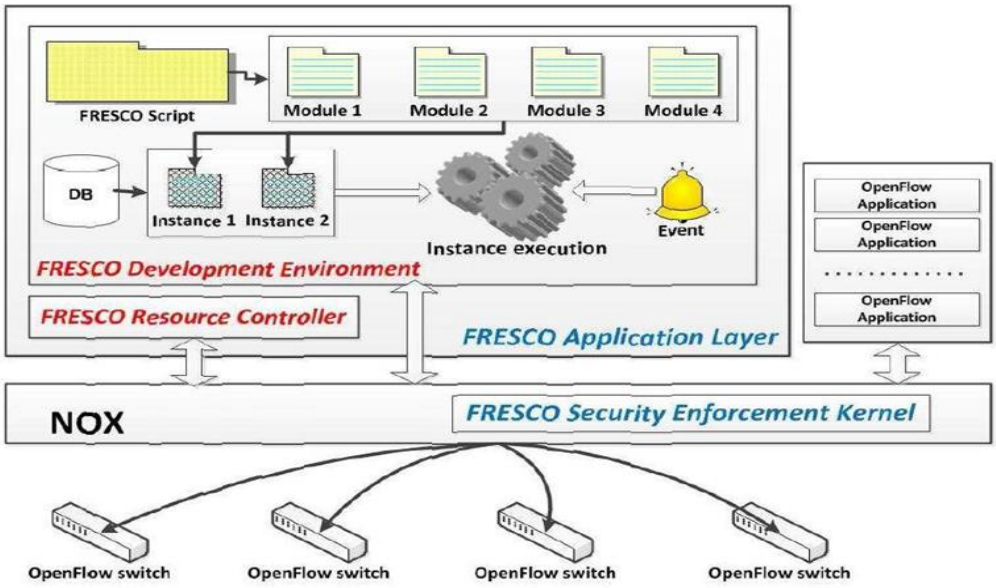
Σχεδιασμός FRESCO

Το FRESCO framework αποτελείται από ένα επίπεδο εφαρμογής και έναν πυρήνα επιβολής ασφαλείας (Security Enforcement Kernel, SEK) ο οποίος ενισχύει τις δράσεις τις πολιτικής απέναντι στις ήδη ανεπτυγμένες εφαρμογές ασφάλειας (Εικόνα 8). Και τα δύο συστατικά είναι ενσωματωμένα στον ελεγκτή NOX.

Το επίπεδο εφαρμογών του FRESCO υλοποιείται χρησιμοποιώντας Python ενότητες του NOX, που επεκτείνονται μέσω APIs του FRESCO για να προσφέρουν δύο βασικές λειτουργίες:

- Περιβάλλον Ανάπτυξης (Development Environment, DE), και
- Ελεγκτή Πόρων (Resource Controller, RC), ο οποίος παρέχει στους προγραμματιστές FRESCO εφαρμογών OpenFlow μεταγωγέα και πρόσβαση,

άγνωστη από τον ελεγκτή, σε γεγονότα και στατιστικά στοιχεία που έχουν να κάνουν με το δίκτυο ροών.

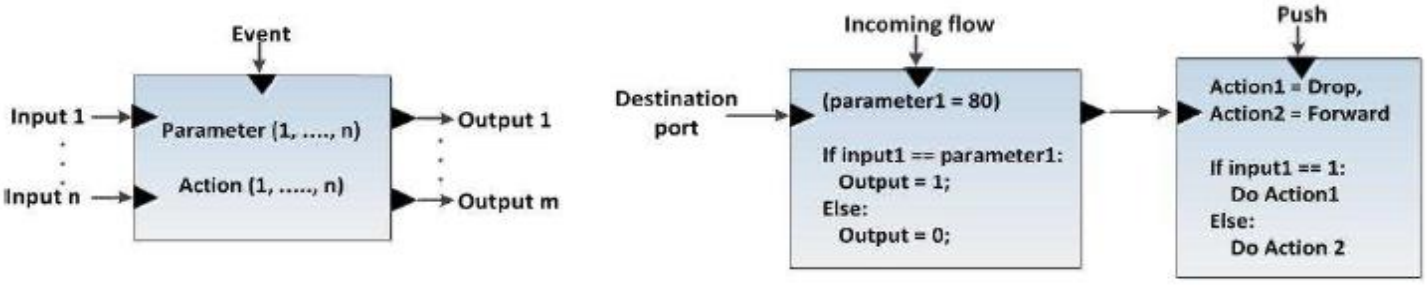


Εικόνα 8: Η αρχιτεκτονική του FRESCO

Για να αρχικοποιηθούν και να οριστούν οι αλληλεπιδράσεις μεταξύ των ενοτήτων ασφαλείας του NOX χρησιμοποιείται η script language του FRESCO. Αυτά τα scripts επικαλούνται εσωτερικά modules , τα οποία αρχικοποιούνται για να σχηματίσουν μια εφαρμογή ασφαλείας η οποία οδηγείται από την είσοδο που καθορίζεται μέσω του FRESCO script και γίνεται προσβάσιμη μέσω του API της βάσης δεδομένων του FRESCO DE. Αυτά τα αρχικοποιημένα modules ενεργοποιούνται (εκτελούνται) από το FRESCO DE καθώς λαμβάνονται triggered γεγονότα εισόδου. Τα FRESCO modules μπορούν επίσης να παράγουν νέους κανόνες ροής, για παράδειγμα, ως απάντηση σε μια απειλή, οι οποίοι στη συνέχεια υποβάλλονται σε επεξεργασία από τον πυρήνα επιβολής ασφαλείας του ελεγκτή.

Η βασική λειτουργική μονάδα στο FRESCO framework είναι το module. Το module αποτελεί το πιο σημαντικό στοιχείο του FRESCO. Όλες οι λειτουργίες ασφαλείας που τρέχουν στο FRESCO πραγματοποιούνται μέσω συναρμολόγησης από modules. Ορίζονται ως αντικείμενα Python και αποτελούνται από 5 τύπους:

- Input,
- Output,
- Parameter,
- Action,
- Event



Εικόνα 9: Σχεδιασμός των modules του FRESCO

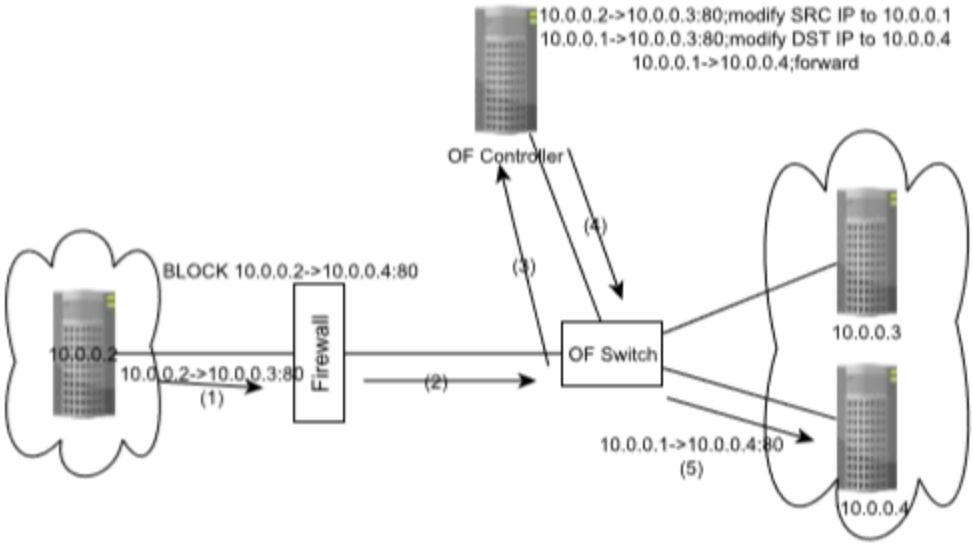
Μια Δράση είναι μια λειτουργία για να διαχειριστούν τα πακέτα του δικτύου (ή ροές). Οι δράσεις που προβλέπονται από FRESCO προέρχονται από τις δράσεις που υποστηρίζονται από τον ελεγκτή NOX OpenFlow. Το πρότυπο OpenFlow καθορίζει τρεις απαιτούμενες δράσεις οι οποίες θα πρέπει να υποστηρίζονται από όλους τους μεταγωγείς του δικτύου OpenFlow:

- Drop, η οποία κάνει ένα πακέτο drop,
- Output, η οποία προωθεί ένα πακέτο σε ένα καθορισμένο τμήμα, και
- Group, η οποία επεξεργάζεται ένα πακέτο μέσω της συγκεκριμένης ομάδας.

Δεδομένου ότι αυτές οι δράσεις πρέπει να υποστηρίζονται από όλους τους μεταγωγείς του δικτύου OpenFlow, το FRESCO τις εξάγει επίσης σε υψηλότερου επιπέδου εφαρμογές.

FRESCO Πυρήνας Επιβολής Ασφάλειας (SEK): Είναι πιθανό οι κανόνες ροής που δημιουργούνται από εφαρμογές που δεν σχετίζονται με την ασφάλεια να έρχονται σε σύγκρουση με τους περιορισμούς ροής που διαμοιράζονται από εφαρμογές FRESCO. Μια σύγκρουση προκύπτει όταν ένας ή περισσότεροι κανόνες ροής επιτρέψουν σε μια ροή να κατευθυνθεί από ένα σημείο σε ένα άλλο που έχει απαγορευτεί ρητά από έναν κανόνα περιορισμού ροής που παράχθηκε από μια εφαρμογή FRESCO.

Οι εφαρμογές OF μπορούν να ανταγωνίζονται, να έρχονται σε αντίθεση, να υπερισχύουν η μια της άλλης, να ενσωματώνουν τρωτά σημεία ή πιθανώς να έχουν γραφτεί από αντιπάλους. Στη χειρότερη περίπτωση ένας αντίπαλος μπορεί να χρησιμοποιήσει την ντετερμινιστική εφαρμογή για τον έλεγχο της κατάστασης όλων των μεταγωγέων στο δίκτυο. Η δυνατότητα να τρέχουν πολλαπλές OpenFlow εφαρμογές (custom και τρίτων) σε μια συσκευή ελέγχου του δικτύου παρουσιάζει μια πρόκληση για την επιβολή της πολιτικής: μιας και διαφορετικές εφαρμογές μπορούν να εισάγουν διαφορετικές πολιτικές ελέγχου δυναμικά, πώς εγγυάται ο ελεγκτής ότι δεν είναι σε σύγκρουση μεταξύ τους; Θεωρήστε ένα απλό παράδειγμα dynamic-flow tunneling, που απεικονίζεται στην εικόνα 10, που περιέχει τρεις hosts, ένα μεταγωγέα και ένα ελεγκτή.



Εικόνα 10: Σενάριο παραδείγματος

Χρησιμοποιείται ένα τείχος προστασίας (το οποίο μπορεί να εφαρμοστεί εύκολα ως εφαρμογή ασφαλείας OpenFlow) που έχει έναν κανόνα για να μπλοκάρει τα πακέτα του δικτύου από τον εξωτερικό host 10.0.0.2 στην υπηρεσία web (θύρα 80) που τρέχει στον εσωτερικό host 10.0.0.4. Τώρα υποθέτουμε ότι κάποια άλλη OpenFlow εφαρμογή προσθέτει τρεις νέους κανόνες ροής στον ελεγκτή.

1. Ο πρώτος κανόνας τροποποιεί τη διεύθυνση IP προέλευσης του πακέτου σε 10.0.0.1 αν ένα πακέτο παραδίδεται από τον 10.0.0.2 στον 10.0.0.3 (θύρα 80).
2. Ο δεύτερος κανόνας αλλάζει τη διεύθυνση IP προορισμού του πακέτου σε 10.0.0.4 αν ένα πακέτο παραδίδεται από τον 10.0.0.1 στον 10.0.0.3 (θύρα 80).
3. Ο τελευταίος κανόνας απλώς επιτρέπει την προώθηση ενός πακέτου από τον 10.0.0.1 στον 10.0.0.4 στη θύρα 80.

Σε αυτή την περίπτωση, αν ο host 10.0.0.2 στέλνει ένα πακέτο στη θύρα 80 του host 10.0.0.3, αυτό το πακέτο μπορεί να παρακάμψει το τείχος προστασίας, επειδή δεν πάει απευθείας στον host 10.0.0.4 αλλά στον 10.0.0.3.
Ωστόσο, αυτό το πακέτο θα παραδοθεί τελικά στον 10.0.0.4 από τον ελεγκτή, ακόμη και αν υπάρχει ένα τείχος προστασίας που απαγορεύει τέτοια κίνηση. Αυτό δείχνει σαφώς ότι μπορεί κανείς να αποφύγει ένα υπάρχον τείχος προστασίας (ή εφαρμογή για την επιβολή ασφαλείας) με την απλή προσθήκη μερικών των κανόνων ροής (από μερικές εφαρμογές). Ενώ αυτό το ενδεικτικό παράδειγμα είναι πολύ απλό, η πραγματική πρόκληση είναι η διασφάλιση ότι όλες οι εφαρμογές του ελεγκτή δεν παραβιάζουν τις πολιτικές ασφαλείας στα δίκτυα του πραγματικού κόσμου (enterprice / cloud) που διαθέτουν πολλούς μεταγωγείς, ποικίλες εφαρμογές και σύνθετες πολιτικές ασφαλείας.

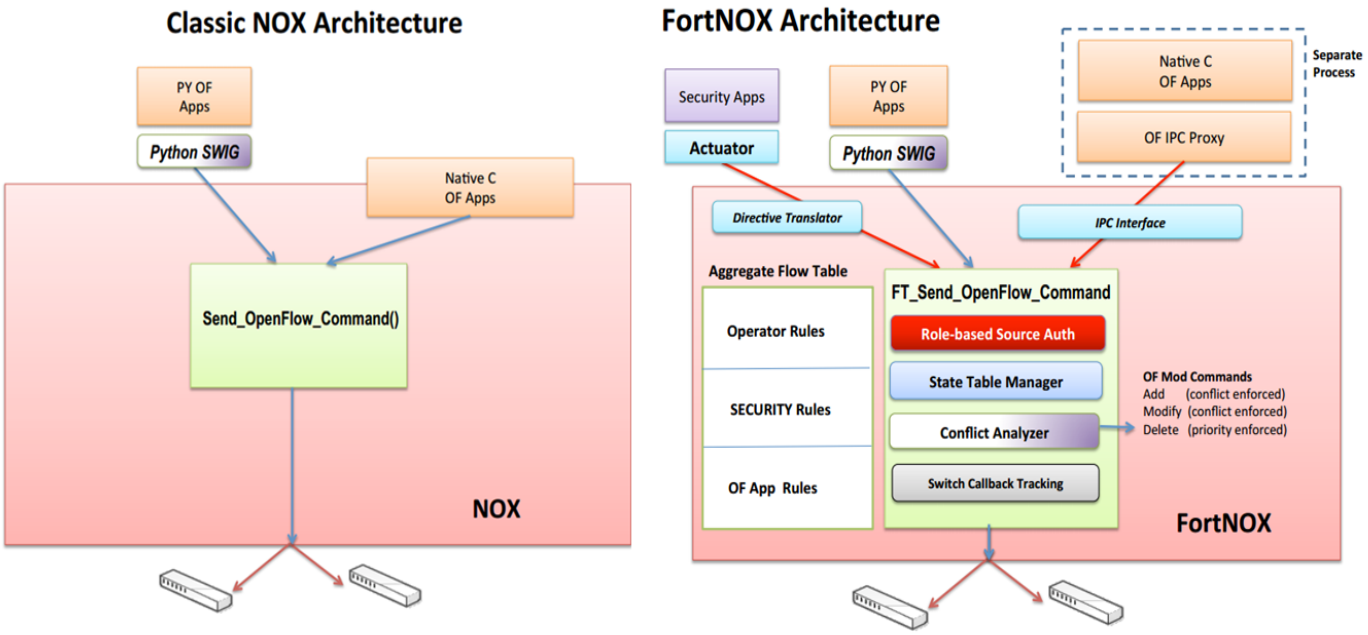
Για τη διαχείριση των περιορισμών ροής και την εκτέλεση αξιολόγησης των συγκρούσεων, το FRESCO ενσωματώνει έναν πυρήνα επιβολής ασφαλείας (SEK), ο

οποίος είναι ενσωματωμένος απευθείας στον ελεγκτή OpenFlow πάνω στον οποίο δραστηριοποιείται το FRESCO. Ο FRESCO SEK προσφέρει αρκετά σημαντικά χαρακτηριστικά βάσει των οποίων το FRESCO εξασφαλίζει ότι οι κανόνες ροής που προέρχονται από τις υπηρεσίες ασφαλείας έχουν προτεραιότητα και εφαρμόζονται έναντι των ανταγωνιστικών κανόνων ροής που παράγονται από εφαρμογές μη σχετιζόμενων με την ασφάλεια:

- Ταυτοποίηση πηγής ενός κανόνα: Ο SEK εισάγει ένα μοντέλο εμπιστοσύνης που επιτρέπει στις εφαρμογές FRESCO να υπογράψουν ψηφιακά κάθε υποψήφιο κανόνα ροής, επιτρέποντας έτσι στον SEK να καθορίζει αν ένας υποψήφιος κανόνας ροής που παράγεται από μια μονάδα ασφαλείας του FRESCO, από μια εφαρμογή OpenFlow, ή από το διαχειριστή του δικτύου.

- Ανίχνευση συγκρούσεων κανόνων: Για την ανίχνευση των συγκρούσεων μεταξύ ενός υποψηφίου κανόνα και το σύνολο των κανόνων που δραστηριοποιούνται επί του παρόντος στο μεταγωγέα, ο SEK ενσωματώνει έναν αλγόριθμο ανάλυσης συγκρούσεων των κανόνων που ονομάζεται alias set rule reduction, ο οποίος ανιχνεύει τις συγκρούσεις κανόνων ροής, συμπεριλαμβανομένων εκείνων που προκύπτουν μέσα από ενέργειες που χρησιμοποιούνται για την παραγωγή εικονικών τούνελ.

- Επίλυση Συγκρούσεων: Όταν προκύπτει μια σύγκρουση ο SEK εφαρμόζει ένα ιεραρχικό μοντέλο εξουσίας. Αυτό το μοντέλο επιτρέπει σε έναν υποψήφιο κανόνα να παρακάμψει (αντικαταστήσει) έναν υπάρχον κανόνα ροής, όταν η ψηφιακή υπογραφή της πηγής του κανόνα θεωρείται ότι διαθέτει περισσότερη εξουσία από την πηγή του κανόνας με τον οποίο είναι σε σύγκρουση.



Εικόνα 11: Αρχιτεκτονική NOX και FortNOX

FortNOX είναι ένας πυρήνας ασφαλείας που επεκτείνει τον ελεγκτή NOX. Παρέχει επιβολή των κανόνων ροής βασισμένη στην πολιτική ασφαλείας και μη παρακαμπτόμενη από τα αιτήματα εισαγωγής νέων κανόνων από τις υπόλοιπες εφαρμογές OpenFlow. Στόχος του είναι να βελτιώσει τον ελεγκτή NOX δίνοντας του τη δυνατότητα να επιβάλει περιορισμούς ροής του δικτύου (που εκφράζονται ως κανόνες ροής). Όταν ένας κανόνας ροής εισάγεται από μια εφαρμογή ασφαλείας στον FortNOX τότε οι υπόλοιπες εφαρμογές δεν μπορούν να εισάγουν κανόνα που να έρχεται σε σύγκρουση με τον πρώτο. Ακόμη, ο FortNOX παρέχει τη δυνατότητα στον διαχειριστή του δικτύου να ορίσει μια αυστηρή πολιτική ασφαλείας που υπερισχύει όλων των δυναμικά καθορισμένων κανόνων ροής.

Επίσης, ενσωματώνει μια live μηχανή ανίχνευσης για σύγκρουση κανόνων, η οποία λειτουργεί σαν διαμεσολαβητής ανάμεσα σε όλες τις αιτήσεις εισαγωγής κανόνων. Μια σύγκρουση κανόνων λέγεται ότι προκύπτει όταν ο υποψήφιος κανόνας OpenFlow ενεργοποιεί ή απενεργοποιεί τη ροή του δικτύου που κανονικά απαγορεύεται (ή επιτρέπεται) από τους υπάρχοντες κανόνες. Η ανάλυση των συγκρούσεων γίνεται χρησιμοποιώντας ένα νέο αλγόριθμο, ο οποίος ονομάζεται alias set rule reduction, ο οποίος εντοπίζει αντιφάσεις μεταξύ των κανόνων. Όταν εντοπίζονται συγκρούσεις, ο FortNOX μπορεί να επιλέξει να αποδεχθεί ή να απορρίψει το νέο κανόνα, ανάλογα με το αν ο αιτών της εισαγωγής λειτουργεί με υψηλότερη άδεια ασφαλείας από εκείνη των συντακτών των κανόνων που έρχεται σε αντιπαράθεση.

Ο FortNOX εφαρμόζει έλεγχο ταυτότητας βάσει ρόλων (role-based authentication) για τον καθορισμό της εξουσιοδότησης ασφάλειας μιας OF εφαρμογής (παραγωγός κανόνα), και επιβάλλει μια αρχή του λιγότερο προνομιούχου για να διασφαλιστεί η ακεραιότητα της διαδικασίας διαμεσολάβησης. Με τον όρο σύγκρουση, αναφερόμαστε σε έναν ή περισσότερους υποψήφιους κανόνες ροής που έχουν σκοπό να καταστήσουν δυνατή μια ροή επικοινωνίας που απαγορεύεται σύμφωνα με τους ισχύοντες κανόνες ροής. Η ικανότητα του FortNOX να προλαμβάνει συγκρούσεις είναι σίγουρα καλύτερη από την απλή ανίχνευση επικαλύψεων, που συνήθως παρέχεται σε μεταγωγείς. Ο FortNOX κατανοεί τις συγκρούσεις μεταξύ των κανόνων ροής, ακόμη και όταν η σύγκρουση αφορά στους κανόνες ροής που χρησιμοποιούν ενέργειες set για να ξαναγράψουν κεφαλίδες πακέτων με τρόπους που εγκαθιδρύουν εικονικό τούνελ μεταξύ δύο τελικών σημείων. Τέλος, ο FortNOX επιλύει τις συγκρούσεις κανόνων που απορρέουν από authorization roles χρησιμοποιώντας ψηφιακά υπογεγραμμένους κανόνες ροής, όπου κάθε εφαρμογή μπορεί να υπογράψει ή όχι κάθε αίτηση εισαγωγής κανόνα ροής, με αποτέλεσμα την εκχώρηση πλεονεκτήματος στον υποψήφιο κανόνα ροής.

Η εικόνα 11 απεικονίζει τα στοιχεία που συνθέτουν την επέκταση FortNOX για τον ελεγκτή NOX. Στο κέντρο του NOX υπάρχει μια διασύνδεση που ονομάζεται `send_openflow_command()`, η οποία είναι υπεύθυνη για τη μετάδοση των κανόνων ροής από μια εφαρμογή στο μεταγωγέα. Ο FortNOX επεκτείνει αυτή τη διασύνδεση με τέσσερις συνιστώσες. Το module που ονομάζεται Rolebased Source Authentication παρέχει ψηφιακή επικύρωση της υπογραφής για κάθε αίτηση εισαγωγής κανόνα ροής, αναθέτοντας την κατάλληλη προτεραιότητα σε έναν υποψήφιο κανόνα ροής ή τη χαμηλότερη προτεραιότητα, εφόσον δεν παρέχεται καμία υπογραφή. Ο Αναλυτής

Συγκρούσεων (Conflict Analyzer) είναι υπεύθυνος για την αξιολόγηση κάθε υποψήφιου κανόνα ροής σε σχέση με το τρέχον σύνολο των κανόνων ροής εντός του συγκεντρωτικού πίνακα ροών. Εάν ο Αναλυτής Συγκρούσεων ορίσει πως ο υποψήφιος κανόνας ροής είναι σύμφωνος με τους ισχύοντες, τότε ο κανόνας διαβιβάζεται στο μεταγωγέα και αποθηκεύεται στον συγκεντρωτικό πίνακα ροών.

Υπάρχουν δύο επιπλέον διεπαφές που επιτρέπουν FortNOX να παρέχει την εφαρμογή διαμεσολάβησης μεταξύ των κανόνων ροής. Πρώτα εισήχθη ένας IPC Proxy που επιτρέπει σε μια OF εφαρμογή σε native C να αρχικοποιηθεί ως μια ξεχωριστή διαδικασία και να διευθύνεται ιδανικά από ένα ξεχωριστό μη προνομιούχο λογαριασμό. Μια διεπαφή προσθέτει μια επέκταση ψηφιακής υπογραφής, επιτρέποντας σε αυτές τις εφαρμογές να υπογράψουν τις αιτήσεις εισαγωγής κανόνων ροής, κάτι το οποίο στη συνέχεια επιτρέπει στο FortNOX να επιβάλλει διαχωρισμούς με βάση τους ρόλους βασιζόμενος σε αυτές τις υπογραφές. Μέσω της διαδικασίας του διαχωρισμού, ο FortNOX είναι σε θέση να επιβάλει μια αρχή «χαμηλότερων προνομίων» στη λειτουργία της υποδομής ελέγχου. Μέσω του μηχανισμού του proxy, οι εφαρμογές OF μπορούν να υποβάλουν νέες αιτήσεις για την εισαγωγή ενός κανόνα ροής, αλλά για αυτά τα αιτήματα η διαμεσολάβηση γίνεται ξεχωριστά και ανεξάρτητα από την υπηρεσία επίλυσης συγκρούσεων που λειτουργεί εντός του ελεγκτή.

Ο FortNOX διαθέτει επίσης έναν μεταφραστή οδηγιών ασφαλείας (security directive translator) που επιτρέπει στις εφαρμογές ασφαλείας να εκφράσουν τις πολιτικές περιορισμού ροών σε ένα υψηλότερο επίπεδο αφαίρεσης, για το οποίο είναι άγνωστη η ύπαρξη του ελεγκτή, το πρωτόκολλο OpenFlow, ή η κατάσταση του μεταγωγέα. Ο μεταφραστής λαμβάνει οδηγίες ασφαλείας από μια εφαρμογή ασφαλείας, στη συνέχεια τις μεταφράζει σε ισχύοντες κανόνες ροής, τους υπογράφει ψηφιακά και τους διαβιβάζει στον FortNOX.

6.6.2 Εφαρμογές ασφάλειας στο SDN

Στην περίπτωση μεγάλων, περίπλοκων, και δυναμικών δικτύων το μεγαλύτερο πρόβλημα είναι πώς χρήστες/ενοικιαστές αναπτύσσουν συσκευές ή λειτουργίες ασφαλείας. Σε μια τέτοια περίπτωση οι ενοικιαστές χρησιμοποιούν μερικές προ-εγκατεστημένες συσκευές ασφαλείας σταθερής θέσης, επειδή δεν είναι σε θέση να συμβαδίσουν με την υψηλή δυναμική της διαμόρφωσης. Οι ενοικιαστές μπορούν επίσης να εγκαταστήσουν συσκευές ασφαλείας για τους ίδιους, αλλά είναι πολύ δύσκολο. Για παράδειγμα, οι Cloud υπηρεσίες είναι πολύ μεγάλες, πολύπλοκες και δυναμικές για αυτό χρειάζεται μια νέα υπηρεσία παρακολούθησης ασφάλειας. Το CloudWatcher είναι μια συσκευή παρακολούθησης της ασφάλειας του δικτύου, που χρησιμοποιεί OpenFlow σε δυναμικά δίκτυα Cloud. Παρέχει παρακολούθηση της ασφάλειας as a Service στο Cloud.

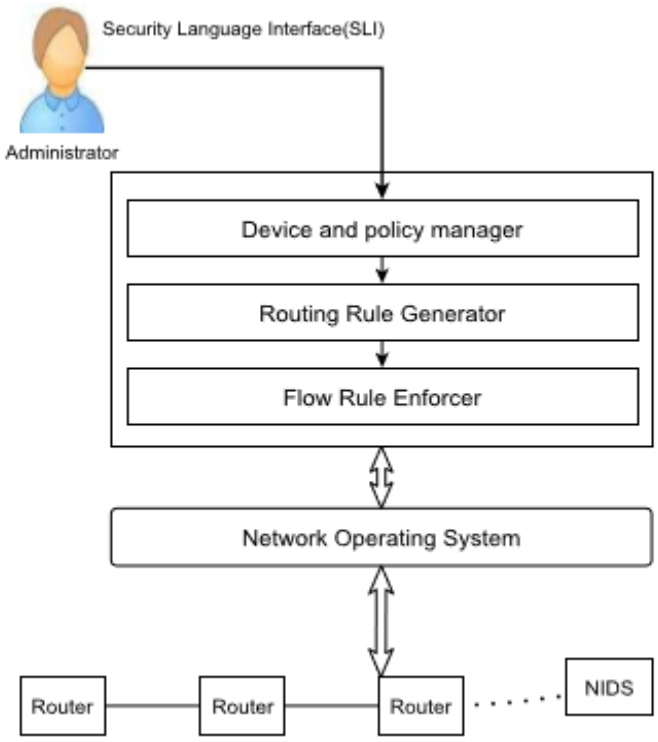
Το CloudWatcher, είναι ένα περιβάλλον μεγάλης κλίμακας που αποτελείται από πολλούς φυσικούς hosts και εικονικές μηχανές (VMs). Για παράδειγμα, το Amazon EC2 Cloud τρέχει τουλάχιστον μισό εκατομμύριο φυσικούς hosts. Κάθε host εξυπηρετεί πολλαπλές

εικονικές μηχανές. Υποθέτοντας ότι κάθε host εξυπηρετεί κατά μέσο όρο δέκα εικονικές μηχανές, το Amazon EC2 Cloud λειτουργεί σχεδόν πέντε εκατομμύρια εικονικές μηχανές. Η διαμόρφωση ενός περιβάλλοντος cloud computing είναι αρκετά περίπλοκη. Για την διαχείριση ενός Cloud δικτύου θα πρέπει να εξεταστεί ο μεγάλος αριθμός των διαφορετικών, δικτυωμένων φυσικών και εικονικών μηχανών και ο μεγάλος αριθμός των διαφορετικών καταναλωτών και ενοικιαστών που μπορεί να απαιτούν πολύ διαφορετικές διαμορφώσεις δικτύωσης. Η διαμόρφωση είναι επίσης αρκετά δυναμική.

Μια από τις ενδιαφέρουσες λειτουργίες του cloud computing είναι οι υπηρεσίες on-demand. Για την προστασία ενός κανονικού δικτύου τύπου Enterprise χρησιμοποιούνται κάποιες συσκευές ασφαλείας δικτύου, όπως firewalls και συστήματα ανίχνευσης εισβολής δικτύου. Αν και είναι δυνατόν να εφαρμοστούν αυτά σε ένα δίκτυο cloud, λόγω των παραπάνω χαρακτηριστικών του cloud computing, υπάρχουν διάφορα θέματα που είναι δύσκολο να αγνοηθούν. Πρώτα θα πρέπει να ληφθούν υπόψιν οι απειλές τόσο από έξω όσο και από μέσα. Οι περισσότερες συσκευές ασφάλειας δικτύου είναι κυρίως εγκατεστημένες σε ένα μέρος όπου ένα δίκτυο είναι συνδεδεμένο με το εξωτερικό, επειδή η επικρατούσα άποψη είναι ότι οι περισσότερες απειλές του δικτύου προέρχονται από εξωτερικά δίκτυα. Ωστόσο, στην περίπτωση ενός δικτύου cloud δεν μπορεί κανείς να βασιστεί εξ ολοκλήρου σε αυτή την υπόθεση. Για παράδειγμα, στην περίπτωση των δημοσίων multi-tenant cloud δικτύων μερικές φορές η ευθύνη των εκτιμήσεων ασφάλειας φορτώνεται στους καταναλωτές / ενοικιαστές τους κάτι που θα μπορούσε να αυξήσει την πιθανότητα προσβολής των εσωτερικών hosts/VMs από κακόβουλο λογισμικό. Σε αυτή την περίπτωση, αν μια εσωτερική VM έχει μολυνθεί θα μπορούσε να μολύνει και τις κοντινές VMs (που μπορεί να ανήκουν σε άλλους καταναλωτές / ενοικιαστές) και δεν θα ανιχνευθεί όταν εγκατασταθούν συσκευές ασφαλείας. Εδώ προκύπτει το ερώτημα του πώς μπορεί κανείς να ανιχνεύσει αυτό το είδος των επιθέσεων. Ένας τρόπος μπορεί να είναι να εγκατασταθούν συσκευές ασφαλείας για κάθε εσωτερικό δίκτυο, για παράδειγμα κατανεμημένα firewalls. Το επόμενο ερώτημα είναι πού θα πρέπει να εγκατασταθούν αυτές συσκευές ασφαλείας. Αφού ένα cloud δίκτυο είναι αρκετά περίπλοκο και δύσκολο να επαναδιαμορφωθεί θα πρέπει να διερευνηθούν προσεκτικά κατάλληλες θέσεις για την εγκατάσταση συσκευών ασφαλείας. Διαφορετικά, μπορεί να χρειαστεί να ρυθμίζονται ή να μετακινούνται συσκευές ασφαλείας πολύ συχνά κάτι που δεν είναι εύκολο στη δεδομένη περίπτωση. Δεύτερον θα πρέπει να αξιοποιηθούν οι συσκευές ασφάλειας του δικτύου λαμβάνοντας υπόψη το δυναμισμό του cloud computing.

Για παράδειγμα, ας υποθέσουμε πως ένα σύστημα ανίχνευσης εισβολής δικτύου έχει εγκατασταθεί σε μια σύνδεση μεταξύ του host A και host B και αφεθεί να παρακολουθεί την κίνηση του δικτύου που παράγεται από μια εικονική μηχανή που υπάρχει στον host A. Ωστόσο, μιας και μιλάμε για εικονικές μηχανές είναι πιθανό η συγκεκριμένη να μετακινηθεί σε έναν άλλο host C. Τότε πρέπει να μεταφερθεί και το σύστημα ανίχνευσης σε μια σύνδεση μεταξύ του host A και host C. Το framework CloudWatcher αντιμετωπίζει αυτά τα ζητήματα και παρέχει τα ακόλουθα πλεονεκτήματα:

- Ελέγχει τις ροές του δικτύου ώστε να εγγυηθεί ότι όλα τα απαραίτητα πακέτα δικτύου επιθεωρούνται από συσκευές ασφαλείας και
- Παρέχει μία απλή script γλώσσα πολιτικής για να βοηθήσει τους ανθρώπους να χρησιμοποιούν τις παρεχόμενες υπηρεσίες εύκολα.



Εικόνα 12: Αρχιτεκτονική του CloudWatcher

Σε σύγκριση με τη διαμόρφωση φυσικών συσκευών, ο έλεγχος των διαδρομών των ροών του δικτύου για να περάσουν από ορισμένους κόμβους του δικτύου είναι πολύ πιο εύκολος από ότι θεωρείται πως είναι. Το SDN με το CloudWatcher παρέχει έναν τρόπο για τον έλεγχο των ροών του δικτύου όπως ο διαχειριστής θέλει. Με τη βοήθεια του SDN το CloudWatcher αλλάζει τις διαδρομές δρομολόγησης για τις ροές του δικτύου και τις κάνει να μεταφέρονται μέσω των κόμβων του δικτύου όπου υπάρχουν οι συσκευές ασφαλείας. Επιπλέον, υποστηρίζει μια απλή script γλώσσα για τη συγγραφή πολιτικών για να επιτρέψει σε έναν διαχειριστή cloud τη χρήση αυτού του framework χωρίς δυσκολία. Είναι αρκετά εύκολο για κάποιον να το μάθει και απλό να το χρησιμοποιήσει.

Αρχιτεκτονική του CloudWatcher: Το CloudWatcher χαρακτηρίζεται ως μια εφαρμογή που τρέχει πάνω σε λειτουργικά συστήματα δικτύου (π.χ. NOX και Beacon), τα οποία χρησιμοποιούνται για τον έλεγχο δρομολογητών ή μεταγωγέων του δικτύου σε περιβάλλοντα SDN. Το CloudWatcher αποτελείται από τρία κύρια μέρη:

1. Διαχείριση συσκευών και πολιτικών που χειρίζονται τις πληροφορίες των συσκευών ασφαλείας.

2. Γεννήτρια κανόνων δρομολόγησης που δημιουργεί κανόνες διαχείρισης πακέτων για κάθε ροή και
3. Όργανο επιβολής των κανόνων ροών που δημιουργήθηκαν σε κάθε μεταγωγέα.

Η συνολική αρχιτεκτονική του CloudWatcher φαίνεται στην εικόνα 12.

7. Mininet

7.1 Το λογισμικό

Το πρόγραμμα Mininet αποτελεί έναν εξομοιωτή δικτύου. Έχει την δυνατότητα να εκτελεί ταυτόχρονα ένα σύνολο από τερματικά, δρομολογητές, μεταγωγείς Ethernet άλλα και των αντίστοιχων συνδέσμων σε ένα ενιαίο Linux Kernel. Χρησιμοποιεί την τεχνολογία της εικονοποίησης ώστε να μπορεί ένα ενιαίο σύστημα να προσομοιώνεται ως ένα πλήρες δίκτυο, χρησιμοποιώντας το ίδιο σύστημα πυρήνα και με τους ίδιους κωδικούς χρήστη. Το κάθε εικονικό τερματικό στο Mininet λειτουργεί σαν ένα πραγματικό τερματικό. Επιπλέον παρέχεται η δυνατότητα ασφαλούς σύνδεσης (τύπου SSH) στο τερματικό, και να εκτελέσει οποιοδήποτε πρόγραμμα (με την προϋπόθεση ότι αυτό είναι εγκατεστημένο στο σύστημα Linux).Τα προγράμματα που εκτελούνται μπορούν να αποστείλουν πακέτα μεταξύ των τερματικών καθώς αναγνωρίζει την σύνδεση μεταξύ τους ως διεπαφές τύπου Ethernet. Η αποστολή των πακέτων πραγματοποιείται με δεδομένη ταχύτητα σύνδεσης και την απαιτούμενη καθυστέρηση. Τα πακέτα επεξεργάζονται από συσκευές που λειτουργούν ως δρομολογητές με δεδομένο χρόνο σε ουρές αναμονής. Όταν δύο προγράμματα επικοινωνούν μέσω Mininet, όπως για παράδειγμα το iperf (το οποίο μετράει την χωρητικότητα της γραμμής μεταξύ δύο σημείων) μεταξύ ενός πελάτη και ενός διακομιστή, η μετρούμενη απόδοση θα πρέπει να είναι κοινή με αυτή των δύο φυσικών μηχανών.

Εν συντομία, στο Mininet, τα τερματικά, οι δρομολογητές, οι μεταγωγείς, οι ελεγκτές και οι συνδέσεις δημιουργούνται με τη χρήση λογισμικού και όχι υλικού. Είναι δυνατή η δημιουργία ενός δικτύου Mininet παρόμοιου με ένα πραγματικό δίκτυο που βασίζεται σε υλικό, ή η δημιουργία ενός υλικού δικτύου παρόμοιου με αυτού του Mininet, τα οποία να εκτελούν τον ίδιο δυαδικό κώδικα και εφαρμογές στην καθεμία πλατφόρμα.

7.2 Πλεονεκτήματα

Όπως γίνεται αντιληπτό το Mininet αποτελεί ένα εύχρηστο και αξιόπιστο εργαλείο στην προσομοίωση δικτύων συγκεντρώνοντας σημαντικά πλεονεκτήματα.

Παρακάτω συνοψίζονται τα πιο αξιοσημείωτα πλεονεκτήματα.

- Η δημιουργία ενός απλουστευμένου δικτύου πραγματοποιείται σε ελάχιστο χρονικό διάστημα με αποτέλεσμα να μπορεί να πραγματοποιηθεί γρήγορα η διαδικασία της αποσφαλμάτωσης.
- Παρέχεται η δυνατότητα εκτέλεσης όλων των λογισμικών που υποστηρίζονται από το λειτουργικό σύστημα Linux.

- Μπορεί να τροποποιηθεί η προώθηση των πακέτων: Οι δρομολογητές Mininet μπορούν να προγραμματιστούν χρησιμοποιώντας OpenFlow πρωτόκολλο.
- Το Mininet μπορεί να εκτελεστεί σε οποιαδήποτε υπολογιστή, διακομιστή, εικονική μηχανή ή ακόμα και σε τεχνολογία τύπου cloud.
- Τα αποτελέσματα του λογισμικού μπορούν να αναπαραχθούν από οποιοδήποτε χρήστη καθώς το μόνο που απαιτείται είναι η εκτέλεση του ιδίου κώδικα στο αντίστοιχο τερματικό.
- Το Mininet αποτελεί εύχρηστο λογισμικό. Για την δημιουργία και την εκτέλεση πειραμάτων απαιτείται προγραμματισμός σε γλώσσα Python.
- Αποτελεί έργο ανοιχτού κώδικα και βρίσκεται υπό ενεργή ανάπτυξη. Η κοινότητα του Mininet αποτελείται από χρήστες και προγραμματιστές και μπορούν να συμβάλλουν στην αντιμετώπιση οποιουδήποτε προβλήματος που μπορεί να αντιμετωπίσει ο εκάστοτε χρήστης.

7.3 Δημιουργία τοπολογιών

Το Mininet υποστηρίζει παραμετροποιήσιμες τοπολογίες. Με την χρήση της γλώσσας προγραμματισμού Python, μπορεί να δημιουργηθεί μια ευέλικτη τοπολογία, βασισμένη στις παραμέτρους που επιθυμεί ο χρήστης, και να χρησιμοποιηθεί ξανά για πολλαπλά πειράματα.

Για παράδειγμα, μια απλουστευμένη τοπολογία δικτύου αποτελούμενη από έναν συγκεκριμένο αριθμό τερματικών συνδεδεμένα σε έναν μεταγωγέα φαίνεται στην εικόνα 13.

```
from mininet.topo import Topo
from mininet.net import Mininet
from mininet.util import dumpNodeConnections
from mininet.log import setLogLevel

class SingleSwitchTopo(Topo):
        "Single switch"
        def build(self, n=2):
                switch = self.addSwitch('s1')
                for h in range(n):
                        host = self.addHost('h%s' % (h+1))
                        self.addLink(host,switch)

        def simpleTest():
                topo = SingleSwitchTopo(n=4)
                net = Mininet(topo)
                net.start()
                print "Dumping"
                dumpNodeConnections(net.hosts)
                print "Testing"
                net.pingAll()
                net.stop()

        if __name__ == '__main__':
                setLogLevel("info")
                simpleTest()
```

Εικόνα 13: Κώδικας για τη δημιουργία τοπολογίας

Σημαντικές κλάσεις, μέθοδοι, συναρτήσεις και μεταβλητές του παραπάνω κώδικα περιλαμβάνουν:

`Topo` : Η βασική κλάση για τις τοπολογίες στο Mininet.

`addSwitch()` : Προσθήκη ενός μεταγωγέα στην τοπολογία και επιστρέφει την ονομασία του.

`addHost()` : Προσθήκη ενός τερματικού στην τοπολογία και επιστρέφει την ονομασία του.

`addLink()` : Προσθήκη ενός αμφίδρομου συνδέσμου στην τοπολογία. Οι σύνδεσμοι στο λογισμικό Mininet είναι αμφίδρομοι, εκτός και αν αναφέρεται διαφορετικά.

`Mininet` : Η βασική κλάση, υπεύθυνη για την δημιουργία και διαχείριση του δικτύου.

`start()` : Ενεργοποίηση λειτουργίας του δικτύου.

`pingAll()` : Έλεγχος συνδεσιμότητας των τερματικών εκτελώντας διαδοχικά ping μεταξύ των κόμβων.

`stop()` : Διακοπή λειτουργίας του δικτύου.

`net.hosts` : Εμφάνιση των ονομασιών όλων των τερματικών στο δίκτυο.

`dumpNodeConnections()` : Απόρριψη συνδέσεων από και προς ένα σύνολο κόμβων.

7.4 Ρύθμιση Παραμέτρων Απόδοσης

Εκτός από την βασική δικτυακή συμπεριφορά, το Mininet παρέχει περιορισμό απόδοσης και χαρακτηριστικά απομόνωσης μέσω των κλάσεων `CPULimitedHost` και `TCLink`.

Μερικές αξιοσημείωτες μέθοδοι και παράμετροι:

`self.addHost(name, cpu=f)` : Επιτρέπει την παραμετροποίηση των πόρων του επεξεργαστή του συστήματος που θα κατανεμηθεί στο εικονικό τερματικό.

`self.addLink( node1, node2, bw=10, delay='5ms', max_queue_size=1000, loss=10, use_htb=True)` : Προσθήκη ενός αμφίδρομου συνδέσμου με χαρακτηριστικά όπως ανεκτικότητα απώλειας, καθυστέρηση και χωρητικότητα, μέγιστος αριθμός πακέτων για ουρά αναμονής τα χίλια πακέτα Η παράμετρος bw εκφράζεται σε Mb/s, η καθυστέρηση σε συμβολοσειρά, η ανεκτικότητα απώλειας σε ποσοστό και το μέγιστο μέγεθος ουράς αναμονής σε πακέτα.

7.5 Μέθοδοι Παραμετροποίησης των Τερματικών

Τα τερματικά του Mininet παρέχουν έναν αριθμό μεθόδων για την παραμετροποίηση του δικτύου.

`IP()` **:** Επιστρέφει την διεύθυνση IP ενός τερματικού ή συγκεκριμένης διεπαφής.

`MAC()` : Επιστρέφει την διεύθυνση MAC ενός τερματικού ή συγκεκριμένης διεπαφής.

`setARP()` : Προσθέτει μια στατική εγγραφή ARP στην κρυφή μνήμη ARP ενός τερματικού.

`setIP()` : Ορίζει την διεύθυνση IP για ένα τερματικό ή μία συγκεκριμένη διεπαφή.

`setMAC()` : Ορίζει την διεύθυνση MAC για ένα τερματικό ή μία συγκεκριμένη διεπαφή.

Για παράδειγμα:

```
print "Host", h1.name, "has IP address", h1.IP(), "and MAC address", h1.MAC()
```

7.6 Μέτρηση Απόδοσης

Σε αυτήν την ενότητα παρουσιάζονται τα σημαντικότερα εργαλεία μέτρησης της απόδοσης του λογισμικού Mininet με τις αντίστοιχες εντολές τους.

- Εύρος ζώνης (`bwm-ng, ethstats`)
- Καθυστέρηση (μέσω της χρήσης της εντολής `ping`)
- Ουρές Αναμονής (χρήση της εντολής `tc` που περιλαμβάνεται στο monitor.py)
- Στατιστικά TCP (`tcp_probe`)
- Χρήση CPU (`top, cpuacct`)

7.7 Πρωτόκολλο OpenFlow και Προσαρμοσμένη Δρομολόγηση

Ένα από τα πιο σημαντικά χαρακτηριστικά του Mininet είναι ότι χρησιμοποιεί το SDN. Χρησιμοποιώντας το πρωτόκολλο OpenFlow και σχετικά εργαλεία, είναι δυνατός ο προγραμματισμός των μεταγωγέων έτσι ώστε να λαμβάνουν αποφάσεις με την λήψη εισερχόμενων πακέτων. Το OpenFlow καθιστά εξομοιωτές όπως το Mininet πολύ χρήσιμους, διότι τα μοντέλα συστήματος δικτύου, συμπεριλαμβανομένου της προσαρμοζόμενης προώθησης πακέτων με την χρήση του OpenFlow, μπορούν εύκολα

να μεταφερθούν σε υλικούς μεταγωγείς OpenFlow για λειτουργίες σχετικά με τον ρυθμό γραμμής.

Ελεγκτές OpenFlow

Εάν εκκινηθεί το Mininet χωρίς να διευκρινίζεται ο ελεγκτής, χρησιμοποιείται από προεπιλογή ο ελεγκτής τύπου ovsc. Η ισοδύναμη εντολή είναι

```
$ sudo mn –controller ovsc
```

Ο συγκεκριμένος τύπος ελεγκτή υλοποιεί έναν απλό μεταγωγέα μάθησης Ethernet, και μπορεί να υποστηρίξει μέχρι 16 μεταγωγείς.

Όταν εκτελεστεί η κλάση Mininet() χωρίς διευκρίνιση κλάσης ελεγκτή, χρησιμοποιείται η προεπιλεγμένη κλάση Controller() για την δημιουργία ενός ελεγκτή τύπου Stanford / OpenFlow. Αντιθέτως, παρέχεται η δυνατότητα χρήσης ενός διαφορετικού τύπου ελεγκτή, δημιουργώντας μια υποκατηγορία του Controller() και μεταφέροντας την στα αρχεία συστήματος του Mininet.

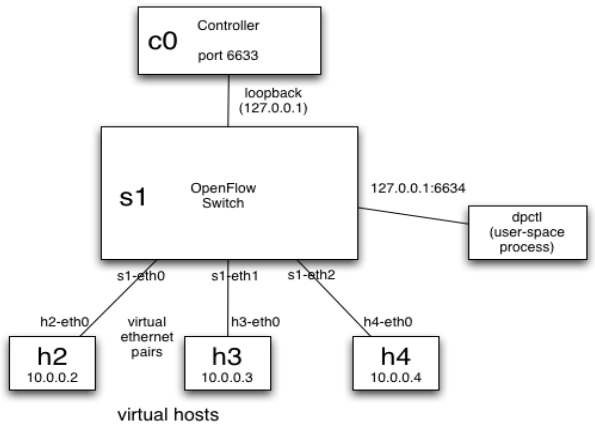
Ακριβείς οδηγίες για την εγκατάσταση του Mininet και των τυχόν επιπλέον προγραμμάτων που χρειάζεται σε κάθε τύπο λειτουργικού συστήματος βρίσκονται στο Παράρτημα 1.

7.8 Δημιουργία Τοπολογίας

Για τη δημιουργία ενός δικτύου με 3 hosts, ένα μεταγωγέα και έναν ελεγκτή OpenFlow χρησιμοποιείται η παρακάτω εντολή

```
$ sudo mn --topo single,3 --mac --switch ovsk --controller remote
```

Αυτή λέει στο Mininet να δημιουργήσει μια τοπολογία με 3 τερματικά και έναν μεταγωγέα τύπου openvSwitch. Δίνει επίσης εντολή η διεύθυνση MAC κάθε τερματικού να είναι ίδια με την διεύθυνση IP του και όλα αυτά να είναι στραμμένα σε έναν απομακρυσμένο ελεγκτή.



Εικόνα 14: Τοπολογία δικτύου

Χρήση του dpctl: είναι ένα βοηθητικό πρόγραμμα που υπάρχει στην διανομή του OpenFlow και επιτρέπει την ορατότητα και τον έλεγχο του πίνακα ροής ενός μεταγωγέα. Είναι ιδιαίτερα χρήσιμο για τον εντοπισμό σφαλμάτων, με την προβολή της κατάσταση ροής και των μετρητών ροής. Οι περισσότεροι μεταγωγείς OpenFlow μπορεί να ξεκινήσουν με μια παθητική θύρα ακρόασης (στις τρέχουσες ρυθμίσεις αυτή είναι η 6634), από την οποία μπορεί να ερευνηθεί ο μεταγωγέας, χωρίς να χρειάζεται να προστεθεί κώδικα εντοπισμού σφαλμάτων στον ελεγκτή.

Με την εισαγωγή της εντολής:

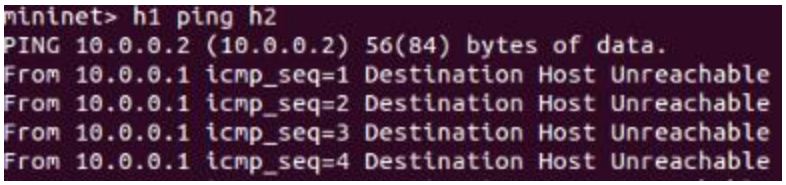
```
$ dpctl dump-flows tcp:127.0.0.1:6634
```

σε ένα δεύτερο SSH terminal γίνεται η σύνδεση με το μεταγωγέα και εμφανίζονται οι δυνατότητες της θύρας καθώς και η κατάσταση της. Μιας και για την ώρα δεν έχει τεθεί σε λειτουργία ο ελεγκτής ο πίνακας ροών είναι άδειος.

Ping Test: Τώρα πίσω στο αρχικό terminal, πληκτρολογώντας την εντολή

```
mininet> h1 ping h2
```

προτρέπουμε τον host 1 να κάνει ping στον host 2. Όμως, καθώς ο πίνακας ροών του μεταγωγέα είναι άδειος, αυτός δεν ξέρει πώς να διαχειριστεί την εισερχόμενη κίνηση και έτσι δεν υπάρχει καμία απάντηση.

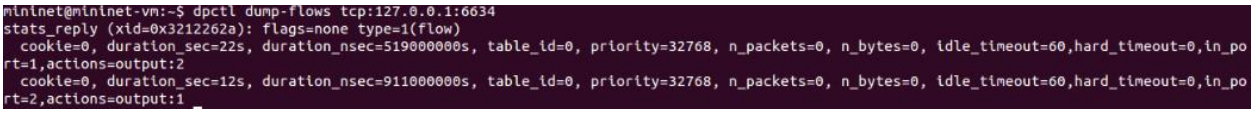


Εικόνα 15:Αδυναμία ping

Στο δεύτερο SSH terminal γίνεται η εισαγωγή των κανόνων ροής χειροκίνητα:

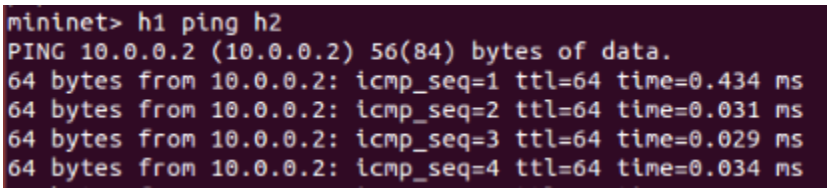
```
$ dpctl add-flow tcp:127.0.0.1:6634 in_port=1,actions=output:2
$ dpctl add-flow tcp:127.0.0.1:6634 in_port=2,actions=output:1
```

Αυτές οι εντολές θα προωθήσουν πακέτα από την θύρα 1 στην 2 και το ανάποδο. Τώρα, με την εισαγωγή της πιο πάνω εντολής παρουσιάζεται ο πίνακας ροών,



Εικόνα 16: Πίνακας ροών

ενώ πλέον η προσπάθεια για ping είναι επιτυχής.

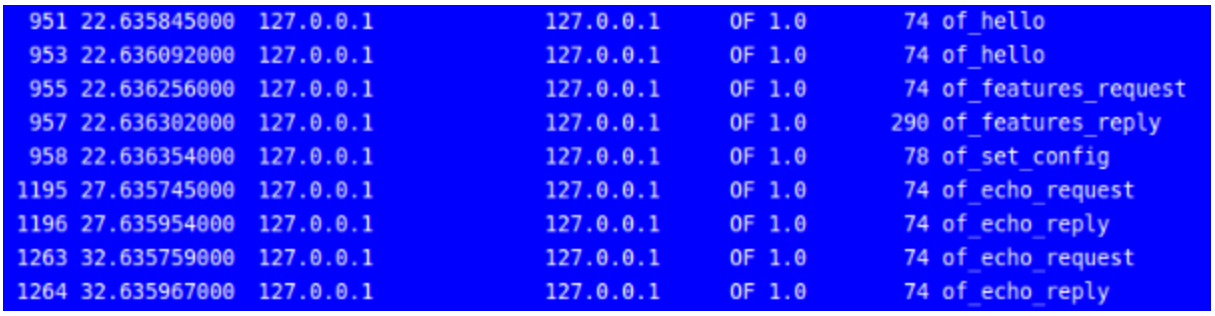


Εικόνα 17:Επιτυχία ping

Έναρξη Wireshark: Το Mininet περιέχει προεγκατεστημένο το Wireshark. Το Wireshark είναι εξαιρετικά χρήσιμο στην παρακολούθηση μηνυμάτων του πρωτόκολλο OpenFlow, καθώς και γενικά τον εντοπισμό σφαλμάτων. Σε ένα νέο SSH terminal με την εισαγωγή της εντολής:

```
$ sudo wireshark &
```

ανοίγει σε νέο παράθυρο το Wireshark. Πατώντας τις επιλογές Capture -> Interfaces -> lo -> Start και εφαρμόζοντας το φίλτρο «of» εμφανίζονται μηνύματα ελέγχου του πρωτοκόλλου OpenFlow.



Εικόνα 18: Έναρξη ελεγκτή

Έναρξη ελεγκτή: Τώρα, με το Wireshark πλέον να «ακούει» την κίνηση σε ένα άλλο SSH terminal ξεκινάει ο ελεγκτής ύστερα από την εντολή:

```
$ controller ptcp:
```

Η εντολή αυτή ξεκινάει έναν απλό ελεγκτή χωρίς εγκατάσταση flow entries. Πλέον στο Wireshark εμφανίζονται πολλά μηνύματα από το αρχικό Hello μεταξύ του ελεγκτή και του μεταγωγέα και μετά. Ο Πίνακας 2 περιέχει τους τύπου μηνυμάτων που ανταλλάσσονται μεταξύ των δύο.

Μήνυμα	Τύπος	Περιγραφή
Hello	Ελεγκτής -> Μεταγωγέας	Όπως και στο TCP handshake ο ελεγκτής στέλνει τον αριθμό της έκδοσης του στον μεταγωγέα
Hello	Μεταγωγέας -> Ελεγκτής	Ο μεταγωγέας απαντά με τον έλεγχο της έκδοσης που υποστηρίζει
Features Request	Ελεγκτής -> Μεταγωγέας	Ο ελεγκτής ζητά να δει ποιες θύρες είναι διαθέσιμες
Set Config	Ελεγκτής -> Μεταγωγέας	Ο ελεγκτής ζητά το μεταγωγέα να στείλει της λήξεις των ροών
Features Reply	Μεταγωγέας -> Ελεγκτής	Ο μεταγωγέας απαντά με μια λίστα από θύρες, ταχύτητα θυρών και υποστηριζόμενους πίνακες και δράσεις
Port Status	Μεταγωγέας -> Ελεγκτής	Επιτρέπει στο μεταγωγέα να πληροφορήσει τον ελεγκτή για αλλαγές σε ταχύτητα θυρών ή συνδεσιμότητα. Αν αγνοηθεί αυτό το μήνυμα θα εμφανιστεί πρόβλημα

Πίνακας 2: Τύποι μηνυμάτων μεταξύ ελεγκτή-μεταγωγέα

Δεδομένου ότι όλα τα μηνύματα που αποστέλλονται μέσω localhost στο Mininet, ο προσδιορισμός του αποστολέα ενός μηνύματος μπορεί να μην είναι εύκολος όταν υπάρχουν πολλοί μεταγωγείς. Ωστόσο, αυτό δεν θα είναι ένα ζήτημα τώρα, δεδομένου ότι υπάρχει μόνο ένας μεταγωγέας. Ο ελεγκτής είναι στην προεπιλεγμένη θύρα OpenFlow (6633), ενώ ο μεταγωγέας είναι σε κάποια άλλη θύρα σε επιπέδου χρήστη.

OpenFlow μηνύματα για Ping: Εφαρμόζοντας το παρακάτω φίλτρο στο Wireshark

```
of && (of.type != 3) && (of.type != 2)
```

και κάνοντας ping από έναν host σε έναν άλλο εμφανίζονται τα μηνύματα που περιγράφονται στον Πίνακα 3.

Μήνυμα	Τύπος	Περιγραφή
Packet – In	Μεταγωγέας -> Ελεγκτής	Ένα πακέτο παρελήφθη και δεν ταιριάζει με καμία εγγραφή στον πίνακα ροών του μεταγωγέα, προκαλώντας το πακέτο να σταλεί στον ελεγκτή
Packet – Out	Ελεγκτής -> Μεταγωγέας	Ο ελεγκτής στέλνει ένα πακέτο σε μία ή περισσότερες θύρες του μεταγωγέα.
Flow – Mod	Ελεγκτής -> Μεταγωγέας	Ο διακόπτης προσθέτει μια συγκεκριμένη ροή στον πίνακα ροών του
Flow Expired	Μεταγωγέας -> Ελεγκτής	Μια ροή έληξε μετά από μια περίοδο αδράνειας

Πίνακας 3: Μυνήματα host-to-host

5424	201.8412740(	10.0.0.1	10.0.0.2	OF 1.0	182 of_packet_in
5425	201.8414130(	127.0.0.1	127.0.0.1	OF 1.0	90 of_packet_out
5427	201.8415820(	10.0.0.2	10.0.0.1	OF 1.0	182 of_packet_in
5428	201.8416440(	127.0.0.1	127.0.0.1	OF 1.0	146 of_flow_add
5443	202.8402940(	10.0.0.1	10.0.0.2	OF 1.0	182 of_packet_in
5444	202.8404310(	127.0.0.1	127.0.0.1	OF 1.0	146 of_flow_add
5473	206.8407060(	00:00:00_00:00:02	00:00:00_00:0(	OF 1.0	126 of_packet_in
5474	206.8409350(	127.0.0.1	127.0.0.1	OF 1.0	146 of_flow_add
5476	206.8412710(	00:00:00_00:00:01	00:00:00_00:0(	OF 1.0	126 of_packet_in

Εικόνα 19:Μηνύματα host

Κατ' αρχάς, φαίνεται ένα ARP request να μην κάνει κάπου match στον πίνακα ροής, κάτι που δημιουργεί ένα broadcast μήνυμα Packet-Out. Στη συνέχεια, η απάντηση ARP έρχεται πίσω. Με γνωστές πλέον διευθύνσεις MAC για τον ελεγκτή, αυτός μπορεί να προωθήσει ροές προς τον μεταγωγέα με ένα μήνυμα Flow-Mod. Μεταγενέστερα ping requests πάνε κατ' ευθείαν μέσα από τη γνωστή διαδρομή, και πλέον δεν υπάρχει λόγος να συμμετέχει ο ελεγκτής στην διαδικασία προώθησης.

Στην δεύτερη φορά που θα γίνει το ping ο χρόνος ολοκλήρωσης του θα είναι μικρότερος γιατί τώρα τα πρώτα μηνύματα δεν θα πάνε στον ελεγκτή.

7.9 Λειτουργία Ελεγκτή

Μέχρι στιγμής έχει χρησιμοποιηθεί ένας απλός ελεγκτής που παρέχεται από προεπιλογή. Τώρα θα τροποποιηθεί ώστε να λειτουργεί ως L2 μεταγωγέας. Σε αυτή την εφαρμογή, ο μεταγωγέας θα εξετάσει κάθε πακέτο και θα κάνει χαρτογράφηση της τοπολογίας μέσω τις θύρας προέλευσης. Στη συνέχεια, η MAC προέλευσης θα συνδέεται με την θύρα. Εάν ο προορισμός του πακέτου έχει ήδη συσχετιστεί με κάποια θύρα, το πακέτο θα σταλεί σε αυτήν, αλλιώς σε όλες τις θύρες του μεταγωγέα.

Υπάρχει ποικιλία ελεγκτών για να διαλέξει κάποιος ανάλογα με τη γλώσσα και τη χρήση που θέλει να κάνει. Στη συγκεκριμένη πτυχιακή χρησιμοποιείται ο ελεγκτής NOX ο οποίος περιλαμβάνει και την εφαρμογή ασφαλείας που θα εφαρμοστεί, την FortNOX.

Όταν ένας μεταγωγέας OpenFlow λαμβάνει ένα πακέτο ενώ δεν υπάρχει ακόμα καταχώρηση για αυτό το πακέτο στον πίνακα ροής, τότε ο μεταγωγέας στέλνει το πακέτο στον ελεγκτή. Στο OpenFlow, κάθε μεταγωγέας (φυσικός ή εικονικός) έχει το δικό του πίνακα ροής. Αυτός ο πίνακας περιέχει πληροφορίες για το πώς το εισερχόμενο πακέτο συμπεριφέρεται μέσα στο δίκτυο. Στη συγκεκριμένη περίπτωση ο πίνακας ροής είναι άδειος. Όταν λοιπόν, για παράδειγμα, ο host-1 προσπαθήσει να επικοινωνήσει με τον host-2 η διαδικασία έχει ως εξής:

a) Ένα πακέτο φτάνει στον μεταγωγέα και δεν υπάρχει καμία εγγραφή στον πίνακα ροών που να του ταιριάζει → το πακέτο θα σταλεί στον ελεγκτή.
b) Ο ελεγκτής εκτελεί τώρα διάφορους ελέγχους για το πακέτο
    i. Γνωρίζει τη διεύθυνση MAC και την θύρα του αποστολέα (host-1); Όχι, πλέον όμως αποθηκεύτηκαν.
    ii. Γνωρίζει τη διεύθυνση MAC και την θύρα του παραλήπτη (host-2); Όχι, προωθεί λοιπόν το πακέτο σε όλες τις ενεργές θύρες.

Ο παραλήπτης (host-2) θα στείλει μια απάντηση στον αποστολέα (host-1), αλλά αυτή τη φορά ο ελεγκτής έχει ήδη κάποιες πληροφορίες.

a) Ο host-2 στέλνει μια απάντηση πίσω στον host-1, αλλά ακόμα δεν υπάρχει εγγραφή στον πίνακα ροής → το πακέτο θα σταλεί στον ελεγκτή για μια ακόμη φορά.
    i. Γνωρίζει τη διεύθυνση MAC και την θύρα του αποστολέα (host-2); Όχι, πλέον όμως αποθηκεύτηκαν.
    ii. Γνωρίζει τη διεύθυνση MAC και την θύρα του παραλήπτη (host-1); Ναι, αποθηκεύτηκαν πριν. Ο ελεγκτής μπορεί να στείλει το πακέτο στη θύρα του παραλήπτη.

Πλέον, οι διευθύνσεις MAC και οι θύρες των δυο συσκευών είναι γνωστές και υπάρχουν εγγραφές για αυτές στον πίνακα ροών. Οι δυο συσκευές μπορούν να επικοινωνούν κατευθείαν χωρίς τη χρήση του ελεγκτή.

7.10 Λειτουργία FortNOX

Η εφαρμογή του Πυρήνα Εφαρμογής Ασφαλείας εκτελείται χρησιμοποιώντας το Mininet. Ο OpenFlow μεταγωγέας διαχειρίζεται την κίνηση μεταξύ ενός demo συστήματος πελάτη-εξυπηρετητή. Υπάρχει ένας ελεγκτής NOX που εισάγει κανόνες ροής στον μεταγωγέα επιτρέποντας στον πελάτη και τον εξυπηρετητή να επικοινωνούν.

Πρώτα θα πρέπει να ξεκινήσει ο ελεγκτής NOX. Με τον NOX να «τρέχει» θα δημιουργηθεί ένας απλός server, ένας client και μια ροή δεδομένων προς και από τον server.

```
10.0.0.4: python tcp_server.py 8888
10.0.0.2: python tcp_client.py 10.0.0.4 8888
```

Αυτή η μικρή εφαρμογή στέλνει και να λαμβάνει 22 bytes από το server στον client.

Τώρα, ας υποθέσουμε ότι η εφαρμογή ασφάλειας αποφασίζει ότι εκείνα τα 22 bytes μεταφέρουν κακόβουλο περιεχόμενο. Λέει στον NOX να στείλει στο μεταγωγέα κανόνες για να μπλοκάρουν την κίνηση. Αυτό γίνεται με την εγκατάσταση ενός τείχους προστασίας ή χρησιμοποιώντας το εργαλείο dpctl. Ο πίνακας ροής μετά από αυτό φαίνεται με το εργαλείο `dump-flow`.

Συνοπτικά, η εφαρμογή ασφαλείας αποφασίζει πως ο πελάτης και ο διακομιστής δεν θα πρέπει να επιτρέπεται να επικοινωνούν και στέλνει δύο κανόνες οι οποίοι μπλοκάρουν την επικοινωνία μεταξύ τους. Για να επιβεβαιωθεί ότι ο κανόνας εφαρμόζεται αρκεί η επανεκκίνηση του server και η προσπάθεια αποστολής κίνησης στον client. Αν όλα έχουν γίνει σωστά αυτό θα οδηγήσει σε timeout, κάτι που επιβεβαιώνει τη σωστή λειτουργία του firewall. Αυτό είναι επαρκές για να δείξει ότι το δίκτυο μπορεί πράγματι να προστατευθεί μέσω της δυναμικής εισαγωγής μιας πολιτικής ασφαλείας (Dynamic Policy Insertion).

Ωστόσο, είναι σημαντικό σε αυτό τη σημείο να τονιστεί ότι το OpenFlow παρέχει πολλά ισχυρά χαρακτηριστικά, όπως την ικανότητα να προγραμματιστεί ο μεταγωγέας για να εκτελέσει επανεγγραφή των περιεχομένων ενός πακέτου συμπεριλαμβανομένων των διευθύνσεων IP. Μια τρίτη εφαρμογή με γνώση προγραμματισμού σε μεταγωγείς μπορεί να επικαλύψει κανόνες μπλοκαρίσματος, ακόμη και αν οι κανόνες αυτοί είναι εγκατεστημένοι με υψηλότερη προτεραιότητα. Αναλυτικότερα:

Ας υποθέσουμε πως ο host-2 στέλνει μια αίτηση σύνδεσης στον host-3. Η εφαρμογή που αναφέρθηκε παραπάνω θα μπορούσε να αντιδράσει με την παροχή virtual tunnel χρησιμοποιώντας κανόνα ροής που συμπεριλαμβάνει ενέργειες τροποποίησης. Οι κανόνες αυτοί, όταν εγκατασταθούν στο μεταγωγέα, συμπεριλαμβάνουν αλλαγή της διεύθυνσης IP αποστολέα από αυτήν του host-2 σε αυτήν του host-5 και την αλλαγή της διεύθυνσης προορισμού από αυτήν του host-3 σε αυτή του host-4.

Ένας συμπληρωματικός κανόνας εισάγεται επίσης για την αντίστροφη ροή. Αυτή η μεταμόρφωση επιτρέπει στην κίνηση του host-2 για να φτάσει στον host-4 δείχνοντας σαν η ροή της κίνησης να είχε ξεκινήσει από τον host-5. Ο κανόνας μπλοκαρίσματος παραμένει ικανοποιημένος, καθώς δεν υπάρχει αίτημα από τον host-2 για να επικοινωνήσει με τον host-4. Ωστόσο, η εικονική ροή μέσω host-3 και host-5 επιτρέπει στον host-2 και τον host-4 να συνεχίσουν την επικοινωνία τους. Αν τώρα προσπαθήσουν ξανά ο server και ο client του παραπάνω παραδείγματος να επικοινωνήσουν (αυτή τη φορά ο client επικοινωνεί με το host-3) αυτό θα γίνει επιτυχώς παρά την πολιτική ασφάλειας η οποία απαγορεύει ρητά να γίνει αυτό.

Για να σταματήσει αυτό το είδος της σύγχυσης ο ελεγκτής NOX υλοποιείται με έναν πυρήνα επιβολής ασφαλείας βάση προτεραιότητας (SEK), τον FortNOX. Μετά την εγκατάσταση του πυρήνα επιβολής ασφαλείας (αρχείο start_forntnox.sh στο directory του NOX) χρειάζεται και εγκατάσταση ξανά των παραπάνω κανόνων μπλοκαρίσματος αλλά και τροποποίησης. Μετά την εγκατάσταση των δεύτερων, ο FortNOX αντιλαμβάνεται τη σύγχυση και απορρίπτει την εγκατάσταση τους.

8. Επίλογος

Η SDN αρχιτεκτονική είναι ακόμα μια νέα μέθοδος δημιουργίας και διαχείρισης δικτύων υπολογιστών, που ακόμη αναπτύσσεται και δοκιμάζεται. Με την πάροδο όμως του χρόνου τα πρωτόκολλα που την υποστηρίζουν (όπως το OpenFlow) γίνονται πιο ώριμα και αξιόπιστα, ενώ αναπτύσσονται όλο και περισσότεροι ελεγκτές σε διάφορες γλώσσες προγραμματισμού, πέραν του NOX, όπως ο Beacon ή ο Floodlight, που προσφέρουν όλο και περισσότερες δυνατότητες για την δημιουργία κεντροποιημένων, λειτουργικών και αξιόπιστων δικτύων υπολογιστών.

Παρά την επιτυχία του Openflow, η ανάπτυξη και η αξιοποίηση πολύπλοκων υπηρεσιών ασφαλείας (εφαρμογών) εξακολουθεί να αποτελεί σημαντική πρόκληση.

Το μέλλον της δικτύωσης υπολογιστών θα βασίζεται όλο και περισσότερο στο λογισμικό, κάτι το οποίο θα επιταχύνει το ρυθμό της καινοτομίας για δίκτυα. Το SDN υπόσχεται να μετατρέψει τα σημερινά στατικά δίκτυα σε ευέλικτες, προγραμματιζόμενες πλατφόρμες με την νοημοσύνη να κατανέμουν τους πόρους δυναμικά, την κλίμακα να υποστηρίξουν τεράστια κέντρα δεδομένων και το εικονοποίηση που απαιτείται για να υποστηρίξουν δυναμικά, ιδιαίτερα αυτοματοποιημένα και ασφαλή περιβάλλοντα cloud.

Πέρα από το SDN την εμφάνιση τους έχουν κάνει και άλλοι όροι όπως SDS (Software Defined Storage) και SDDC (Software Defined Data Center). Οι όροι αυτοί είναι μέρος μιας ευρύτερης τάσης που ονομάζεται Software Defined Everything (SDE), η οποία έχει ως στόχο η υπολογιστική υποδομή να εικονοποιείται και να παραδίδεται ως υπηρεσία.

Ο κόσμος της πληροφορικής σίγουρα κινείται προς ένα μέλλον καθοριζόμενο από λογισμικό. Παρόλα αυτά, η μετάβαση μπορεί να πάρει χρόνια για έναν οργανισμό, αν ποτέ μπορεί να γίνει πλήρως.

Με τα πολλά πλεονεκτήματά του και την εκπληκτικά πρόθυμη βιομηχανία, το SDN είναι στο δρόμο για να γίνει το νέο πρότυπο για τα δίκτυα.

Βιβλιογραφία

1. https://www.sdxcentral.com/resources/security/what-is-software-defined-security/
2. https://www.opennetworking.org/
3. http://mininet.org/
4. http://blog.silver-peak.com/six-benefits-of-sdn
5. http://www.infocom.gr/wp-content/uploads/2015/10/nfv-2-10-15-sent.pdf
6. http://etherealmind.com/basics-explaining-the-difference-between-software-defined-networking-sdn-and-network-functions-virtualization-nfv/
7. https://www.sdxcentral.com/resources/sdn/what-the-definition-of-software-defined-networking-sdn/
8. http://www.nojitter.com/post/240169834/4-challenges-lying-in-the-wait-of-sdn
9. http://www.ciena.com/connect/blog/4-Reasons-SDN-is-Secure.html
10. https://networkingexchangeblog.att.com/enterprise-business/security-benefits-software-defined-networking-sdn/#fbid=CUq9XS3YfkV
11. Βουκελάτου Ουρανία Στέλλα Σ., *Ελεγκτής OpenFlow για Υπηρεσίες με Επίγνωση Περιβάλλοντος*, 2015, Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
12. Ρεντίφης Χριστόφορος, *Ανάπτυξη λογισμικού με χρήση ελεγκτή OpenFlow και υλοποίηση μηχανισμών δρομολόγησης πακέτων*, 2014, Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
13. IRTF, RFC-7426, 2015
14. Metzler Jim, Metzler Ashton & Associates, *An Overview of Openflow*, 2014, Webtorials
15. Metzler Jim, Metzler Ashton & Associates, *The 2015 Guide to SDN and NFV*, 2015, Webtorials
16. ONF, *Relationship of SDN and NFV*, 2015
17. ONF, *SDN Architechture*, Issue 1.1, 2016
18. ONF, *Software-Defined Networking: The New Norm for Networks*, 2012
19. Prasad Biswajit, *Security Issues in Software Defined Networking*, 2014, Jadavpur University, Department of Computer Science and Engineering

20. Scott-Hayward Sarah, O'Callaghan Gemma, Sezer Sakir, *SDN Security: A Survey*, Queen's University Belfast
21. Smeliansky R.L., *SDN for Network Security*, 2012, Moscow State University
22. Taylor Martin, *A guide to SDN and NFV*, 2014, Metascitch Networks
23. Taha Abdalla, *Software-Defined Networking and its Security*, 2014, Aalto University, School of Electrical Engineering

Παράρτημα

Εγκατάσταση Mininet

- **Προαπαιτούμενα**

Χρειάζεται ένας υπολογιστής με το λιγότερο 1GB μνήμης RAM και τουλάχιστον 5GB κενού χώρου. Όσο πιο γρήγορος ο επεξεργαστής του συστήματος τόσο πιο γρήγορος θα είναι και ο χρόνος που θα κάνει boot η εικονική μηχανή. Οι παρακάτω οδηγίες απευθύνονται σε λογισμικά Windows, Linux και Mac. Τα Linux και Mac προτιμώνται επειδή απαιτούνται λιγότερα στην εγκατάσταση. Επίσης, πρέπει να υπάρχει πρόσβαση διαχειριστή στο μηχάνημα.

- **Προαπαιτούμενο Λογισμικό**

Κάθε αρχείο θα χρειαστεί να κατέβει ξεχωριστά. Τα αρχεία περιλαμβάνουν λογισμικό εικονοποίησης, ένα SSH terminal, έναν X server και μια εικόνα εικονικής μηχανής.

Το VirtualBox επιτρέπει την εκτέλεση μιας εικονικής μηχανής μέσα σε μια φυσική μηχανή, και είναι δωρεάν και διαθέσιμο για Windows, Mac και Linux. Μπορείτε να εξάγετε την εικόνα VirtualBox σε μορφή VMDK και να το χρησιμοποιήσετε με VMWare χρησιμοποιώντας τις παρακάτω οδηγίες.

Οι ακόλουθες οδηγίες προϋποθέτουν τη χρήση του VirtualBox, αλλά οι οδηγίες εφαρμόζονται ανεξάρτητα από το εικονικό λογισμικό αφού ολοκληρωθεί η αρχική ρύθμιση.

Αρχεία για λήψη

Για τη λήψη της συμπιεσμένης VM εικόνας:

- Virtual Machine εικόνα (μορφή OVF, 64-bit, Mininet 2.2.0) (Συνιστάται για όλους τους σύγχρονους υπολογιστές και λειτουργικά συστήματα)
- Virtual Machine εικόνα (μορφή OVF, 32-bit, Mininet 2.2.0) (Συνιστάται για παλαιό hardware και Windows)

Από τη διεύθυνση https://github.com/mininet/openflow-tutorial/wiki/Installing-Required-Software

Σημαντικό: Για αυτές τις εικόνες VM, το όνομα χρήστη είναι «mininet» με κωδικό «mininet».

Η μορφή OVF μπορεί να εισαχθεί στο VirtualBox, VMware, ή άλλα δημοφιλή προγράμματα εικονικοποίησης. Για τα υπόλοιπα αρχεία που θα χρειαστούν ανάλογα με το λειτουργικό σύστημα:

Τύπος	Έκδοση	Λογισμικό Εικονοποίησης	X Server	Terminal
Windows	7+	Virtual Box	Xming	PuTTY
Windows	XP	Virtual Box	Xming	PuTTY
Mac	OS X 10.7-10.9 Lion/Mountain Lion/ Mavericks	Virtual Box	XQuartz	Terminal (built in)
Mac	OS X 10.5-10.6 Leopard/Snow Leopard	Virtual Box	X11 ή XQuartz	Terminal (built in)
Linux	Ubuntu 10.4+	Virtual Box	X server (built in)	Gnome terminal (built in)

Πίνακας 4: Προγράμματα για εγκατάσταση

- **Εγκατάσταση**

Μετά τη λήψη του κατάλληλου λογισμικού και εικόνας VM, βεβαιωθείτε ότι κάθε στοιχείο της στήλης (X server, λογισμικό εικονοποίησης, και SSH terminal) στην επιλεγμένη πλατφόρμα έχει εγκατασταθεί και λειτουργεί, και ότι η εικόνα VM και τρέχει σωστά.

1. Εισαγωγή εικόνας: Αφού έχετε κατεβάσει το .ovf εικόνα, ξεκινήστε το VirtualBox, στη συνέχεια, επιλέξτε File> Import Appliance και επιλέξτε την εικόνα .ovf που έχετε κατεβάσει.

2. VM Setup: Επιλέξτε το VM και πηγαίνετε στην καρτέλα Ρυθμίσεις. Πηγαίνετε στο Network-> Adapter 2. Επιλέξτε το πλαίσιο «Enable Adapter», και μετά από το drop down menu επιλέξτε το «host-only adapter». Αυτό επιτρέπει την εύκολη πρόσβαση στην εικονική μηχανή από το τερματικό σας. Σε αυτό το σημείο είναι όλα έτοιμα να ξεκινήσει η εικονική μηχανή με το εικονίδιο "Start" ή διπλό κλικ. Στο παράθυρο της κονσόλας VM, συνδεθείτε με το όνομα χρήστη και τον κωδικό πρόσβασης που είναι και στις δύο περιπτώσεις "mininet". Σημειώστε ότι αυτός ο χρήστης είναι sudoer, ώστε να μπορείτε να εκτελέσετε εντολές με δικαιώματα root.

3. Πρόσβαση στο δίκτυο: Για να μην αυξηθεί πολύ το μέγεθος της εικόνας το Mininet δεν έχει περιβάλλον εργασίας. Τα σενάρια πραγματοποιούνται μέσω X forwarding, όπου τα γραφικά εμφανίζονται μέσω ενός X server που τρέχει στο φυσικό μηχάνημα. Για να ξεκινήσει η διαδικασία X forwarding πρέπει να βρεθεί η διεύθυνση IP του εικονικού μηχανήματος.

   Στο Virtual Box πρέπει η εικόνα που θα φορτωθεί να έχει δύο διεπαφές δικτύου. Η μία θα είναι μια διεπαφή NAT για να παρέχει πρόσβαση στο Internet και η άλλη

πρέπει να είναι μια host-only διεπαφή που επιτρέπει την επικοινωνία με το φυσικό μηχάνημα. Για παράδειγμα, η διεπαφή NAT θα έχει IP διεύθυνση 10.x ενώ η host-only διεπαφή θα έχει διεύθυνση 192.168.x. Από ένα terminal γίνεται σύνδεση SSH στην host-only διεπαφή. Και οι δύο διεπαφές διαμορφώνονται με DHCP.

4. Πρόσβαση στο VM μέσω SSH: Σε αυτό το σημείο γίνεται η σύνδεση μεταξύ του φυσικού μηχανήματος και του εικονικού μέσω SSH. Αφού τρέξει το εικονικό μηχάνημα και γίνει η σύνδεση (username και password είναι mininet), πληκτρολογήστε την εντολή:

```
ifconfig –a
```

Θα πρέπει να εμφανιστούν τρείς διεπαφές (eth0, eth1, lo). Αν κάποια από τις διεπαφές δεν έχει IP διεύθυνση τότε χρησιμοποιήστε την εντολή

```
sudo dhclient ethX
```

Μετά από αυτό η σύνδεση εξαρτάται από το λειτουργικό σύστημα του εικονικού μηχανήματος.

- MAC και Linux:

  ```
  ssh –X mininet@[host-only if IP]
  ```

  Εισάγετε τον κωδικό ξανά και μετά πληκτρολογήστε την εντολή

  ```
  xterm
  ```

  Ένα καινούριο terminal πρέπει να εμφανιστεί και αυτό ήταν, η σύνδεση έγινε!

- Windows

  Για να χρησιμοποιήσετε τις εφαρμογές X11, όπως το xterm και το Wireshark, ο διακομιστής Xming πρέπει να εκτελείται, και θα πρέπει να κάνετε μια σύνδεση ssh με ενεργοποιημένη τη διαδικασία X11 forwarding.

  Κατ 'αρχάς, ξεκινήστε Xming. Δεν εμφανίζεται παράθυρο, αλλά μπορείτε να βεβαιωθείτε ότι λειτουργεί με την εύρεση της διαδικασία του στο Windows Task Manager.

Δεύτερον, κάντε μια σύνδεση ssh με ενεργοποιημένο το X11 forwarding. Ανοίξτε την εφαρμογή puTTY, και συνδεθείτε εισάγοντας τη διεύθυνση IP της host-only διεπαφής.

Για να ενεργοποιήσετε το X11 forwarding από το GUI του puTTY, κάντε κλικ

puTTY-> Connection -> SSH-> X11,

στη συνέχεια, κάντε κλικ στο

Forwarding -> "Enable X11 Forwarding.