HELLENIC REPUBLIC
National and Kapodistrian University of Athens
EST. 1837

DEPARTMENT OF INFORMATICS & TELECOMMUNICATIONS

README

Ομάδα 5

Ηλίας Καλαμάτας 1115201400053
Θεοδώρα Παναγέα 1115201400135
Φωτεινή Τσάβο 1115201500206

Σεπτέμβριος 2020

Περιεχόμενα

1 Σχετικά με το project

Η ομάδα μας αποτελείται από τρία άτομα και η απόφαση για το ποιο project θα επιλέξουμε από τα δύο, ήταν ομόφωνη. Προτιμήθηκε το πρώτο project (**Mininet-Wifi / OpenDayLight**), διότι κατά τη διάρκεια της έρευνάς μας βρήκαμε αρκετό υλικό, όπου μπορέσαμε να βασιστούμε για να υλοποιήσουμε ενδιαφέρουσες εφαρμογές.

1.1 Εγκατάσταση

Το στάδιο της εγκατάστασης αποδείχθηκε αρκετά χρονοβόρο, καθώς λόγω τεχνικών προβλημάτων δοκιμάσε το κάθε μέλος της ομάδας να εγκαταστήσει τα εργαλεία με διαφορετικούς τρόπους. Έτσι πετύχαμε και τον πειραματισμό μας σε διαφορετικά περιβάλλοντα, ώστε να διαπιστώσουμε τη λειτουργικότητα των εργαλείων μας.

- Το πρώτο μέλος αρχικά δοκίμασε την εγκατάσταση των εργαλείων σε Virtual Machines, μιας και το μόνο λειτουργικό σύστημα του υπολογιστή ήταν τα Windows 10. Χρησιμοποιώντας λοιπόν το Oracle Virtual Box εγκατέστησε δύο VMs· ένα για το OpenDayLight και ένα για το Mininet-WiFi. Παρατηρώντας ότι η ταχύτητα εκτέλεσης των VMs δεν ήταν ιδιαίτερα optimal, αποφάσισε να καταφύγει σε dual-boot με Ubuntu 16.04 LTS. Στη συνέχεια ακολούθησε όλες τις οδηγίες εγκατάστασης σύμφωνα με το documentation του e-class για όλες τις εγκαταστάσεις. (Mininet-WiFi, ODL Boron SR4, Java 8).

- Το δεύτερο μέλος δοκίμασε την εγκατάσταση σε Ubuntu 20.04 LTS και μετά σε 18.04 LTS, όπως αναφερόταν στις οδηγίες των διαφανειών. Επίσης, σε 2ο υπολογιστή και περιβάλλον macOS, εγκαταστάθηκε το Mininet-Wifi σε VM με εικόνα που βρήκαμε από εδώ και τοπικά, το OpenDayLight έκδοσης Boron SR4.

- Το τρίτο μέλος πραγματοποίησε την εγκατάσταση και των δύο εργαλείων Mininet-Wifi, OpenDayLight σε δύο ξεχωριστά Virtual Machines, τα οποία υποστηρίζουν Ubuntu 18.04 LTS. Για να επιτευχθεί η γρήγορη λειτουργία και να μην επιβαρυνθεί το σύστημα, χρειάστηκε να μειωθεί η base memory σχεδόν στο ελάχιστο που απαιτούσε το κάθε εργαλείο.

Τα features που εγκαταστήσαμε είναι τα εξής:

- **odl-dlux-all:** OpenDaylight graphical user interface
- **odl-l2switch[1]-switch:** Provides network functionality similar to an Ethernet switch
- **odl-restconf:** Allows access to RESTCONF API
- **odl-mdsal-apidocs:** Allows access to Yang API

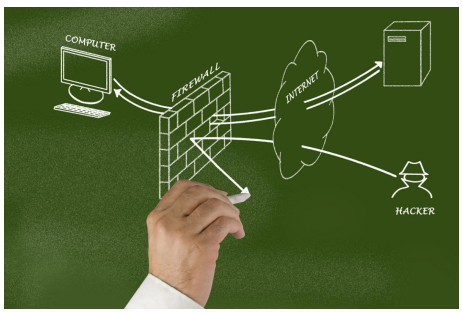
Επιπλέον, τα εργαλεία που πρέπει να υπάρχουν στον υπολογιστή για την εκτέλεση της εργασίας μας, είναι:

1. Το πακέτο pandas της python που χρησιμοποιούμε στην εραρμογή Firewall και το οποίο εγκαθίσταται με την εντολή python -m pip install pandas.
2. Το πρόγραμμα VLC που χρησιμοποιείται στην εφαρμογή Handover για πραγματοποίηση του Video Streaming.

1.2 Επιλογή εφαρμογών

- **Firewall:** Έχουμε ένα δίκτυο, όπου στην αρχή υπάρχει επικοινωνία μεταξύ όλων των nodes, και στη συνέχεια ξεκινάμε ένα firewall σε κάποιο access point και προσθέτουμε κανόνες, ώστε να περιορίσουμε την επικοινωνία.

- **Handover:** Έχουμε 2 stations(sta1 και sta2), που είναι συνδεδεμένα σε διαφορετικά access points, τοποθετημένα αντιδιαμετρικά και ξεκινούν σε διαφορετικούς χρόνους να κινούνται. Ο "sta1" ξεκινάει video streaming προς τον "sta2" και κατά την κίνησή τους, γίνεται handover σε 3 access points.



[1] Υπεύθυνο και για την αυτόματη δημιουργία και ανανέωση του πίνακα των flows του δικτύου. Να σημειωθεί, ότι δοκιμάστηκε η δημιουργία τους χειροκίνητα, αλλά το feature παρέμεινε για διευκόλυνση.

2 Tasks

2.1 Γενικά

Αρχικά, συζητήσαμε τα προβλήματα που αντιμετώπιζε ο καθένας με την εγκατάσταση των εργαλείων. Αφού τα λύσαμε, προχωρήσαμε στο επόμενο στάδιο, με σκοπό τη δημιουργία δικής μας τοπολογίας. Παρατηρήθηκε πως σε δύο από τα τρία μέλη της ομάδας, δεν εμφανίζονταν τα επιθυμητά αποτελέσματα στο DLUX, καθώς και ότι υπήρχε πρόβλημα με τη δημιουργία flows στο ODL, αντίθετα με το πρώτο μέλος. Επομένως, το μεγαλύτερο μέρος της εργασίας εκπονήθηκε ομαδικά, από τη στιγμή που τα προγράμματά μας ήταν εκτελέσιμα μόνο σε έναν υπολογιστή.

Πρώτα, πειραματιστήκαμε με διάφορες τοπολογίες, μέχρι να καταλήξουμε σε αυτές που χρησιμοποιήσαμε στις εφαρμογές μας. Στη συνέχεια, κάναμε διάφορες δοκιμές όσον αφορά τα flows και είδαμε το αντίκτυπο που είχαν αυτές στη συμπεριφορά του δικτύου. Επειδή η εφαρμογή του Firewall μάς φάνηκε πιο απαιτητική, ξεκινήσαμε από αυτήν, μεγαλύτερο μέρος της οποίας υλοποιήθηκε ομαδικά. Μετά την ολοκλήρωση του Firewall, αποφασίσαμε να ασχοληθούμε με το Handover, όπου ψάξαμε πώς μπορούμε να δείξουμε τη λειτουργία της μέσω video streaming.

2.2 Tasks ανά άτομο

Αν και όπως αναφέρθηκε, το μεγαλύτερο μέρος της εργασίας έγινε ομαδικά, υπήρχαν κάποια tasks τα οποία υλοποιήθηκαν ατομικά.

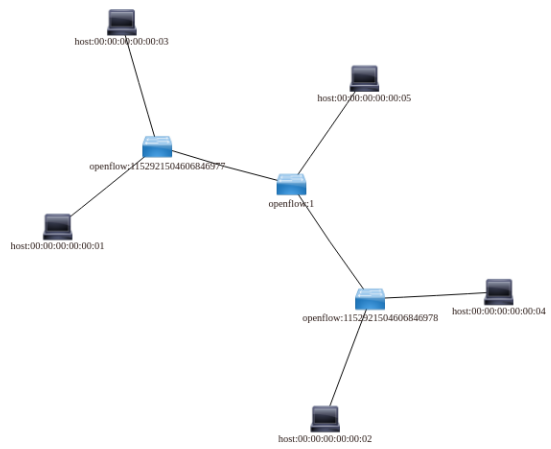
- Θεοδώρα: Υλοποίηση συνάρτησης για τη συλλογή κανόνων του Firewall, όπου κάθε ένας αποτελεί διαφορετική οντότητα σε μία δομή δεδομένων dictionary.
- Ηλίας: Χρήση της παραπάνω συνάρτησης για έλεγχο διπλότυπου κανόνα πριν την προσθήκη του στον πίνακα κανόνων του Firewall και την εφαρμογή του.
- Φωτεινή: Δοκιμή/Σύνδεση/Χρήση του VLC Media Player για την πραγματοποίηση live video streaming μεταξύ των stations της τοπολογίας.

3 Firewall

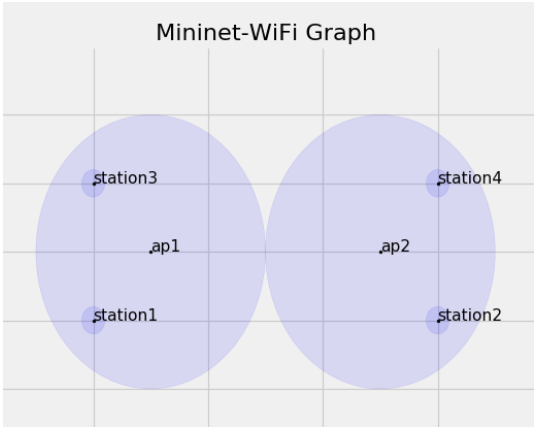
3.1 Εισαγωγή

Ένα "τοίχος προστασίας" είναι ένα πρόγραμμα ρυθμισμένο με τέτοιο τρόπο, ώστε να επιτρέπει ή να απορρίπτει συγκεκριμένα πακέτα δεδομένων που μεταφέρονται από ένα δίκτυο υπολογιστών σε ένα άλλο. Μπορεί να κατηγοριοποιηθεί είτε ως Software Firewall είτε ως Hardware Firewall. Ένα SDN-based-Firewall διαφέρει σε αρκετά σημεία από ένα παραδοσιακό, όπως για παράδειγμα ότι ο έλεγχος πραγματοποιείται κεντρικά και συγκεκριμένα στον Controller.

3.2 Τοπολογία [2]



Σχήμα 1: Τοπολογία από το DLUX



Σχήμα 2: Mininet-WiFi Graph

[2] Οποιοδήποτε παραπάνω node στο DLUX σε σχέση με το graph του Mininet είναι host και οποιοδήποτε παραπάνω bridge είναι switch.

3.3 Περιγραφή

Η εφαρμογή εκτελεί τις εξής λειτουργίες:

1. Δημιουργία της παραπάνω τοπολογίας, η οποία πιο συγκεκριμένα, αποτελείται από 4 Stations, 2 Access Points, 1 Switch και 1 Host.
2. Έλεγχος επικοινωνίας μεταξύ των παραπάνω nodes (Ping Reachability).
3. Εμφάνιση των components του δικτύου, καθώς και MAC και IP διευθύνσεις για καθένα από αυτά.
4. Εκκίνηση του Firewall σε ένα από τα Access Points.
5. Εισαγωγή κανόνων για την αποδοχή/απόρριψη πακέτων.

Κατά την εκκίνηση του Firewall, δίνεται το εξής μενού επιλογών:

- Εμφάνιση πίνακα υπάρχοντων κανόνων του Firewall. (Αρχικά ο πίνακας είναι κενός.)
- Εφαρμογή Basic κανόνα. (Οι κανόνες αυτοί είναι συγκεκριμένοι και ήδη φτιαγμένοι-μή τροποποιήσιμοι.)
- Εφαρμογή Custom κανόνα. (Ευχέρεια του χρήστη για τη δημιουργία ενός δικού του κανόνα.)

3.4 Υλοποίηση

Στο φάκελο Firewall υπάρχουν τα εξής αρχεία:

- **fw_main.py:** Αποτελεί το κύριο κομμάτι κώδικα, που συνδέει και εκτελεί όλες τις λειτουργίες της εφαρμογής.
- **fw_topo.py:** Περιλαμβάνει την κλάση, που δημιουργεί την τοπολογία για την ανάδειξη της εφαρμογής, καθώς και διάφορες μεθόδους, που χρησιμοποιούνται από την τοπολογία, όπως για παράδειγμα η "print_elements()", η οποία εκτυπώνει όλα τα στοιχεία του δικτύου και επιπλέον πληροφορίες για αυτά.
- **fw_app.py:** Σε αυτό το αρχείο υλοποιήθηκαν τα εξής:
  1. Η κλάση **Firewall**, η οποία χρησιμοποιεί την κλάση **Rule** για να δημιουργήσει και να εφαρμόσει κανόνες, οι οποίοι απορρίπτουν ή δέχονται πακέτα κατά την επικοινωνία του δικτύου, και τη συνάρτηση **flows_dict()** για να πάρει από τον controller μια λίστα με flows, που είναι στην ουσία οι παραπάνω κανόνες, που βρίσκονται σε εφαρμογή στο Firewall, και πραγματοποιεί την εκτύπωση τους.
  2. Η συνάρτηση **Run_Firewall()** η οποία συνθέτει τα παραπάνω, δημιουργεί το μενού του Firewall και το θέτει σε λειτουργία.
- **fw_api.py:** Περιέχει μία κλάση υπεύθυνη για την απευθείας επικοινωνία του προγράμματος με το Rest API του Controller και την επιστροφή του πίνακα με τα flows.

3.5 Εκτέλεση

Για την εκτέλεση αυτής της εφαρμογής ακολουθούμε τα εξής:

- Τρέχουμε την εντολή sudo python fw_main.py
- Αφού ολοκληρωθεί η δημιουργία της τοπολογίας, βρισκόμαστε πλέον στο κεντρικό μενού.
- Πληκτρολογούμε "ping" για να δοκιμάσουμε την επικοινωνία μεταξύ των nodes. Παρατηρούμε πως όλα τα στοιχεία του δικτύου επικοινωνούν επιτυχώς μεταξύ τους.
- Πληκτρολογούμε "nodes" για να δούμε τα components του δικτύου. Εμφανίζεται η επιλογή για εμφάνιση περισσότερων στοιχείων των nodes (MAC + IP address). Πληκτρολογούμε YES ή NO. Να σημειωθεί πως στο πεδίο controller φαίνεται η IP του. Αν επιθυμούμε μπορούμε πληκτρολογώντας την επιλογή "CLI" στο κεντρικό μενού και με την εντολή "sh ovs-vsctl list controller" να επιβεβαιώσουμε, πως ο ενεργός controller είναι όντως ο ODL.
- Πληκτρολογώντας "firewall" μας δίνεται η επιλογή να διαλέξουμε bridge και πραγματοποιείται η εκκίνησή του σε αυτό. (Διαλέγουμε την επιλογή 2 για εκκίνηση στο ap1.)

- Ξεκινώντας το Firewall μεταφερόμαστε σε ένα καινούργιο μενού. Με την εντολή "help" εμφανίζονται οι διαθέσιμες εντολές, ενώ με την εντολή "exit" επιστρέφουμε στο προηγούμενο μενού. Συγκεκριμένα με την εντολή "help" εμφανίζονται οι εξής εντολές:

  1. SR: Προβολή πίνακα κανόνων σε ισχύ. (Show Rules)
  2. CBR Δημιουργία βασικού κανόνα. (Create Basic Rule)
  3. CCR Δημιουργία προσαρμοσμένου κανόνα. (Create Custom Rule)

- Διαλέγουμε την επιλογή CBR και από τους βασικούς κανόνες διαλέγουμε τον πρώτο κανονα. (Block incoming ping requests.) Με την εντολή SR μπορούμε να σιγουρευτούμε, ότι ο νέος κανόνας βρίσκεται πλέον σε ισχύ.
- Κάνουμε έξοδο από το Firewall και δοκιμάζουμε πάλι την επικοινωνία του δικτύου με την εντολή ping.[3]

Πατήστε εδώ για να παρακολουθήσετε ένα βίντεο με ενδεικτική εκτέλεση όλων των παραπάνω λειτουργιών. Λόγω αυτού δεν συμπεριλήφθηκαν και στιγμιότυπα οθόνης της εκτέλεσης της εφαρμογής.

3.6 Συμπεράσματα

Παρατηρούμε τα εξής:

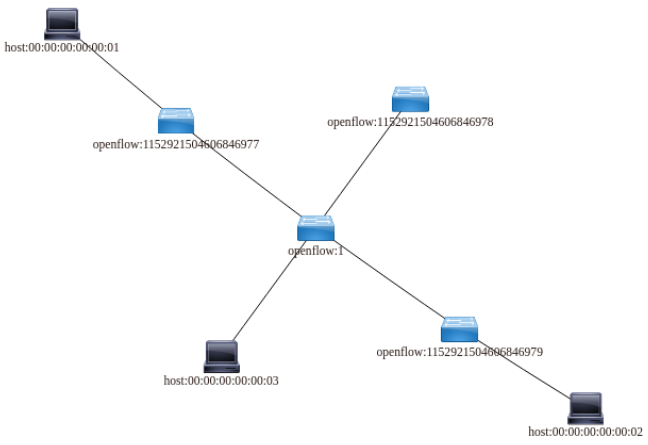
- Στην αρχή η επικοινωνία μπορεί να πραγματοποιηθεί μεταξύ όλων των στοιχείων του δικτύου.
- Εφαρμόζουμε στο ap1 τον κανόνα για να μπλοκάρουμε τη μεταφορά των ICMP πακέτων.
- Έτσι πλέον τέτοιου είδους επικοινωνία μεταξύ των group (sta1,sta3) και (sta2,sta4,host5) αποκόπτεται.
- Η επικοινωνία, που παραμένει μεταξύ των sta1 και sta3, ενδεχομένως μπορεί να δικαιολογηθεί από το γεγονός, ότι μόλις εγκαθιδρυθεί η επικοινωνία τους, μπορούν να επικοινωνήσουν απευθείας, δίχως να χρειάζεται η μεσολάβηση του Access Point. (AdHoc Networking)
- Επομένως, έστω ότι ο host5 αποτελεί εισβολέα στο δίκτυο με σκοπό να πραγματοποιήσει IP Spoofing. Συμπεραίνουμε πως σε μια τέτοια περίπτωση, το Firewall προστατεύει τα (sta1,sta3), ενώ τα (sta2,sta4) βρίσκονται εκτεθιμένα στον κίνδυνο.

4 Handover

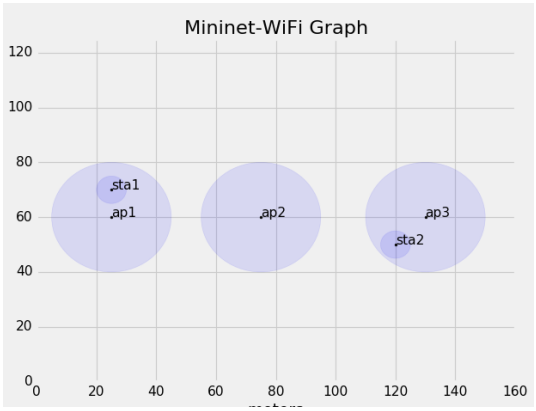
4.1 Εισαγωγή

Στις Τηλεπικοινωνίες ο όρος Handover/Handoff αναφέρεται στη διαδικασία μεταφοράς μιας τρέχουσας κλήσης ή μεταφοράς δεδομένων από ένα κανάλι σε ένα άλλο.

4.2 Τοπολογία [4]



Σχήμα 1: Τοπολογία από το DLUX



Σχήμα 2: Mininet-WiFi Graph

[3] Επιπλέον, πληκτρολογώντας την επιλογή "api" στο κεντρικό μενού, έχουμε τη δυνατότητα να δούμε όλα τα flows του controller σε κάποιο bridge για να εξακριβώσουμε ότι οι κανόνες του firewall έχουν εισαχθεί επιτυχώς.

[4] Οποιοδήποτε παραπάνω node στο DLUX σε σχέση με το graph του Mininet είναι host και οποιοδήποτε παραπάνω bridge είναι switch.

4.3 Περιγραφή

Η εφαρμογή δημιουργεί την παραπάνω τοπολογία, η οποία πιο συγκεκριμένα αποτελείται από 2 Stations, 3 Access Points, 1 Switch και 1 Host. Κατά την εκκίνηση της, δίνεται και η επιλογή ύπαρξης ή μη κενού μεταξύ των εμβελειών των Access Points. Στη συνέχεια, τα Stations πραγματοποιούν κίνηση στο χώρο, ούτως ώστε να περάσουν διαδοχικά στην εμβέλεια των διαφορετικών Access Points του δικτύου. Οι αρχικές θέσεις των δύο stations φαίνονται στο graph της τοπολογίας. Πρώτα ξεκινάει την κίνηση του ο sta1 κινούμενος από το ap1 στο ap2 και τερματίζει την διαδρομή του, φτάνοντας στο ap3. Ύστερα ο sta2 ξεκινάει την ακριβώς αντίθετη διαδρομή και τερματίζει στο ap1.
Για τον έλεγχο επιτυχούς πραγματοποίησης του Handover χρησιποιούνται 2 διαφορετικές μέθοδοι:

- Ping μεταξύ ενός Host, που βρίσκεται συνδεδεμένος στο δίκτυο, και των Stations. Παρατηρείται πως όταν ένα από τα 2 Stations βρεθεί ανάμεσα από τις εμβέλειες των Access Points, στιγμιαία χάνονται λίγα πακέτα και μόλις βρεθεί πάλι εντός εμβέλειας η επικοινωνία αποκαθίσταται, καθώς το Handover πραγματοποιήθηκε επιτυχώς.
- Video Streaming με τη χρήση του VLC από το sta1 προς τον sta2. Αντίστοιχα, η ροή του βίντεο διακόπτεται στιγμιαία όταν κάποιο από τα Stations βρεθεί εκτός εμβέλειας, όμως λόγω του επιτυχημένου Handover, η επικοινωνία αποκαθίσταται.

4.4 Υλοποίηση

Στο φάκελο Handover υπάρχουν τα εξής αρχεία:

- **ho_main.py:** Αποτελεί το κύριο κομμάτι κώδικα, που συνδέει και εκτελεί τις λειτουργίες της εφαρμογής.
- **ho_topo.py:** Περιλαμβάνει την κλάση, που δημιουργεί την τοπολογία για την ανάδειξη της εφαρμογής, καθώς και τη μέθοδο "stream()", που χρησιμοποιείται από την τοπολογία για την πραγματοποίηση ζωντανής μετάδοσης βίντεο.

4.5 Εκτέλεση

Για την εκτέλεση αυτής της εφαρμογής ακολουθούμε τα εξής:

- Τρέχουμε την εντολή sudo python ho_main.py
- Εμφανίζεται μήνυμα στην οθόνη σχετικά με την ύπαρξη ή μη κενού ανάμεσα στα Access Points. Πληκτρολογούμε YES ή NO για να κάνουμε την αντίστοιχη επιλογή.
- Αφου ολοκληρωθεί η δημιουργία της τοπολογίας δίνεται ένα χρονικό περιθώριο 30 δευτερολέπτων στο χρήστη για την επιλογή της δοκιμής του. (Μετά από αυτό ξεκινάει η κίνηση των stations και πραγματοποιείται το Handover). Οι διαθέσιμες επιλογές για τη δοκιμή είναι ping και video streaming.
- Για την πραγματοποίηση ελέγχου επικοινωνίας των nodes (ping) πληκτρολογούμε "CLI". Στην συνέχεια εκτελούμε την εντολή "h3 ping sta1" και περιμένουμε να δούμε πώς η κίνηση του sta1 θα επηρεάσει την επικοινωνία του με τον host3. Στην συνέχεια όταν ο sta1 φτάσει στο ap3, τερματίζουμε το ping (CTRL+C), καθώς ο sta1 μένει πλέον στάσιμος μέσα στην εμβέλεια του Access Point. Επαναλαμβάνουμε την ίδια διαδικασία για τον sta2, ο οποίος θα ξεκινήσει σύντομα την κίνηση του κι αυτός προς το ap1.
- Όταν ολοκληρωθούν και οι δύο διαδρομές μπορούμε να τερματίσουμε το πρόγραμμα. (έξοδος από το CLI με "exit", έξοδος από το κύριο μενού με "exit")
- Τρέχουμε την εφαρμογή άλλη μια φορά για να κάνουμε δοκιμή με το video streaming. (Διαδικασία: sudo python ho_main.py > YES/NO > video)
- Θα πρέπει να εμφανιστούν στην οθόνη δύο παράθυρα του VLC. Στο παράθυρο με όνομα "VLC Media Player", πατάμε το κουμπί του PLAY, που βρίσκεται στο κάτω μέρος του παραθύρου. Πατάμε "Add..." και διαλέγουμε το βίντεο[5], που θέλουμε να μεταδώσουμε. Στη συνέχεια πατάμε Open και Play. Η μετάδοση του βίντεο θα πρέπει τώρα να ξεκινήσει κανονικά στο δεύτερο παράθυρο, που άνοιξε προηγουμένως.

Πατήστε εδώ και εδώ για να παρακολουθήσετε βίντεο με ενδεικτική εκτέλεση όλων των παραπάνω λειτουργιών. Λόγω αυτού δεν συμπεριλήφθηκαν και στιγμιότυπα οθόνης της εκτέλεσης της εφαρμογής.

[5] Στο φάκελο του Handover, υπάρχει ενδεικτικό βίντεο για χρήση με όνομα owl_yeet_kitten.mp4

4.6 Συμπεράσματα

Παρατηρούμε τα εξής:

- Όταν κάποιο από τα δύο Stations βρεθεί εκτός εμβέλειας, στιγμιαία έχουμε απώλεια κάποιων πακέτων κατά την πραγματοποίηση του Handover, είτε λόγω π.χ. της διαδικασίας εναλλαγής των Access Points, είτε λόγω π.χ. του validation, που χρειάζεται κατά την είσοδο του στο νέο Access Point. (Το τελευταίο θα μπορούσε να βελτιωθεί με τη χρήση του Fast Roaming - IEEE-802.11r)
- Η απώλεια πακέτων είναι αντιληπτή είτε υπάρχει κενό μεταξύ των Access Points είτε όχι. Απλώς στην πρώτη περίπτωση το ενδεχόμενο να χαθούν περισσότερα πακέτα είναι μεγαλύτερο. Επιπλέον υπάρχει και η πιθανότητα αποκοπής της επικοινωνίας, αναλόγως το μέγεθους του κενού.
- Η απώλεια πακέτων κατά τη διάρκεια του Video Streaming μεταφράζεται ως πάγωμα της εικόνας - διακοπή ήχου στη μεριά του παρατηρητή.

5 Βιβλιογραφία

Ενδεικτικά αναγράφονται οι παρακάτω πηγές, που μελετήθηκαν.

1. https://usermanual.wiki/Pdf/mininetwifidraftmanual.297704656/view/
2. https://www.opensourceforu.com/2016/07/implementing-a-software-defined-network-sdn-based-firewall/
3. https://www.ijsdr.org/papers/IJSDR1805076.pdf
4. https://journalijcar.org/sites/default/files/issue-files/5603-A-2018.pdf
5. https://commotionwireless.net/es/docs/cck/networking/types-of-wireless-networks/
6. https://www.frank-durr.de/?p=68
7. https://wiki.videolan.org/Documentation:StreamingHowTo/StreamaFile/
8. https://wiki.videolan.org/Documentation:Commandline/
9. http://csie.nqu.edu.tw/smallko/sdn/vlc.htm?
10. http://mininet.org/api/classmininet11node11Node.html
11. https://www.brianlinkletter.com/mininet-wifi-software-defined-network-emulator-supports-wifi-networks/
12. https://mininet-wifi.github.io/mobility/
13. https://mininet-wifi.github.io/propagation/
14. http://www.openvswitch.org//support/dist-docs/ovs-ofctl.8.txt
15. http://trainer.edu.mirantis.com/SDN50/ovs.html