ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ

ΔΙΔΑΚΤΙΚΗ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΚΑΙ ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

Μεταπτυχιακή Διπλωματική Εργασία

ΜΕΛΕΤΗ ΚΑΙ ΑΝΑΛΥΣΗ ΕΠΙΔΟΣΗΣ ΤΟΥ MULTIPATH TCP ΣΕ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ ΕΠΙΚΟΙΝΩΝΙΑΣ

ΔΗΜΗΤΡΗΣ Ι. ΒΑΣΙΛΕΙΑΔΗΣ

Επιβλέπων Καθηγητής: Ευθύμογλου Γεώργιος, Επίκουρος Καθηγητής

Πειραιάς, Ιούνιος 2012

ΠΕΡΙΛΗΨΗ

Τα κινητά τερματικά τελευταίας τεχνολογίας, όπως είναι τα smartphones και τα tablets, διαθέτουν πολλαπλές ασύρματες διεπαφές, κάθε μια από αυτές με διαφορετικές ιδιότητες, όπως διαφορετικό εύρος ραδιοκάλυψης, διαφορετική ταχύτητα μεταφοράς δεδομένων και διαφορετική κατανάλωση ενέργειας. Ωστόσο, καμία από αυτές τις συσκευές δεν διαθέτει την δυνατότητα ταυτόχρονης αξιοποίησης των διεπαφών αυτών, κάτι που θα ήταν χρήσιμο στον χρήστη σε αρκετές περιπτώσεις, με κυριότερες αυτές που συνδέονται με την κινητικότητά του. Η μετάβαση από την μία ζεύξη στην άλλη επιφέρει τελικά διακοπή της ροής των δεδομένων, με αποτέλεσμα να απαιτείται η εγκατάσταση νέας σύνδεσης.

Η παρούσα διπλωματική εργασία παρουσιάζει την ανάλυση του Multipath TCP, ενός νέου πρωτοκόλλου που ανήκει στο επίπεδο μεταφοράς, που έχει ως στόχο την εκπομπή και λήψη δεδομένων σε δύο ή περισσότερες ζεύξεις ταυτόχρονα εντός μιας σύνδεσης. Στην εργασία παρουσιάζονται τα αποτελέσματα των δοκιμών που διενεργήθηκαν, τόσο σε επίπεδο προσομοίωσης, χρησιμοποιώντας τον προσομοιωτή ns-2, όσο και σε πραγματικές συνθήκες, χρησιμοποιώντας δύο είδη διεπαφών: WiFi και 3G/HSPA. Για τις δοκιμές σε πραγματικές συνθήκες, χρησιμοποιήθηκε η υλοποίηση του Multipath TCP στο λειτουργικό σύστημα Linux, που δημιουργήθηκε από τον Sebastian Barré, του Πανεπιστημίου της Λέουβεν στο Βέλγιο (Université catholique de Louvain - Belgique). Τα αποτελέσματα δείχνουν ότι το πρωτόκολλο

επιτυγχάνει παραπλήσια ταχύτητα μεταφοράς δεδομένων σε τερματικά με δύο διεπαφές WiFi, σε σχέση με το άθροισμα των ταχυτήτων μεταφοράς όταν χρησιμοποιούνται οι δύο διεπαφές ξεχωριστά. Επίσης τα αποτελέσματα δείχνουν την αδυναμία χρήσης του πρωτοκόλλου σε δίκτυα 3G/HSPA, λόγω της αφαίρεσης των TCP Options, που ορίζει το πρωτόκολλο και είναι απαραίτητα για την εγκατάσταση των υποροών, από τους ενδιάμεσους κόμβους (middleboxes) του κορμού δικτύου από την πλευρά του 3G Παρόχου. Στην περίπτωση αυτή, παρόλη την μη επίτευξη μιας σύνδεσης Multipath TCP, η σύνδεση συνεχίζει ως μια κανονική TCP σύνδεση, χρησιμοποιώντας μόνο μια εκ των δύο διεπαφών.

Λέξεις – κλειδιά: Multihoming, Multipath TCP, MPTCP

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

BGP	Border Gateway Protocol
DNS	Domain Name System
DSN	Data Sequence Number
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISN	Initial Sequence Number
ISP	Internet Service Provider
MPTCP	Multi-Path Transmission Control Protocol
MSS	Maximum Segment Size
PEP	Performance Enhancing Proxy
PLR	Packet Loss Ratio
RFC	Request For Comments
RIR	Regional Internet Registry
RTT	Round Trip Time

SCTP	Stream Control Transmission Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
ΑΣ	Αυτόνομο Σύστημα
ΔΕΑ	Διαμεσολαβητής Ενίσχυσης Απόδοσης
ΠΜΔ	Περιφερειακό Μητρώο Διαδικτύου
ΠΥΔ	Πάροχος Υπηρεσιών Διαδικτύου

ΑΠΟΔΟΣΗ ΟΡΩΝ

END-USER	Τελικός Χρήστης
FIREWALL	Τείχος Προστασίας
MIDDLEBOX	Ενδιάμεσος Κόμβος
MULTIPATH TCP	Πρωτόκολλο Ελέγχου Πολυδιαδρομικής Μετάδοσης
PROXY	Διαμεσολαβητής
ROUTER	Δρομολογητής

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Συστήματα Πολλαπλών Συνδέσεων (Multihoming)

Ο πυρήνας του Διαδικτύου διαχειρίζεται από πολλά και διάφορα εμπλεκόμενα μέρη. Εντούτοις, όλοι συμμερίζονται τον στόχο για την επίτευξη σύνδεσης μεταξύ των μερών με την μέγιστη αποδοτικότητα, αξιοπιστία και με το ελάχιστο κόστος. Ο συνδετικός κρίκος, ο οποίο καθιστά το γεγονός αυτό εφικτό είναι το πρωτόκολλο BGP (Border Gateway Protocol). Στην ορολογία του πρωτοκόλλου BGP, οι εταίροι καλούνται Αυτόνομα Συστήματα (Autonomous Systems). Τα ΑΣ ορίζονται ως ένα σύνολο από δικτυακούς πόρους, όπως για παράδειγμα δρομολογητές, ζεύξεις, κ.ά., οι οποίοι συμμερίζονται μια ενιαία πολιτική δρομολόγησης. Το πρωτόκολλο BGP επιτρέπει στα ΑΣ να παρέχουν υπηρεσίες διαβίβασης σε άλλους επί πληρωμή ή να συνάψουν συμφωνία αμοιβαίας διαβίβασης χωρίς κόστος. Η τελευταία περίπτωση καλείται επιχειρηματική σχέση ανταλλαγής κίνησης [Gao01]. Η οικειοθελής διασύνδεση (peering) αναφέρεται στην σχέση μεταξύ Αυτόνομων Συστημάτων, όπου οι εταίροι συμφωνούν στην ανταλλαγή κίνησης από τους αντίστοιχους πελάτες χωρίς κάποιο οικονομικό αντίτιμο.

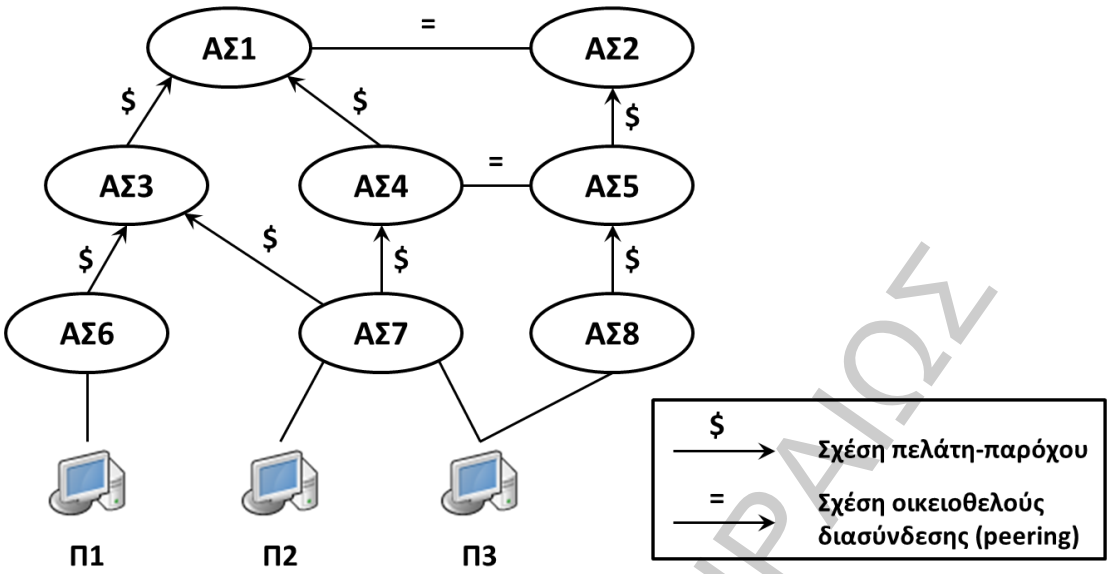
Στην περίπτωση όπου ένα ΑΣ (πάροχος) πληρώνεται για την προσφορά υπηρεσίας ανταλλαγής κίνησης σε ένα άλλο ΑΣ (πελάτης), τότε η συμφωνία αναφέρεται ως σχέση παρόχου - πελάτη [Gao01]. Στο *Σχήμα 1* απεικονίζεται ένα απλοποιημένο Διαδίκτυο, στο οποίο μερικά Αυτόνομα Σύστημα (ΑΣ6, 7, 8) παρέχουν συνδεσιμότητα στο Διαδίκτυο στους πελάτες τους (Π1, Π2, Π3). Αυτά τα ΑΣ βρίσκονται στις άκρες του Διαδικτύου και καλούνται *στελέχη*. Αφ' ετέρου, τα ΑΣ που προσφέρονται μόνο υπηρεσίες μεταφοράς σε άλλα ΑΣ καλούνται *ΑΣ διέλευσης*, και βρίσκονται στον πυρήνα του Διαδικτύου. Ο όρος Πάροχος Υπηρεσιών Διαδικτύου (Internet Services Provider - ISP), που θα χρησιμοποιηθεί στη συνέχεια, αναφέρεται στα Αυτόνομα Συστήματα (στελέχη ή ΑΣ διέλευσης) που παρέχει συνδεσιμότητα με το Διαδίκτυο σε πελάτες (τελικός χρήστης ή ακόμα και σε άλλο Πάροχο).

Ένας πελάτης, για την επιλογή του Παρόχου Υπηρεσιών Διαδικτύου, θα επηρεαστεί συνήθως από το κόστος και την ποιότητα (αξιοπιστία, επιδόσεις) της παρεχόμενης υπηρεσίας. Οι απαιτήσεις αυτές μεταφράζονται για τον ΠΥΔ ως η ενέργεια ελαχιστοποίησης κόστους σε συνδυασμό με την μεγιστοποίηση της αξιοπιστίας και της αποτελεσματικότητας της μεταφοράς δεδομένων. Στο Σχήμα 1, το ΑΣ6 είναι ευάλωτο σε μια αποτυχία του ΑΣ3. Ακόμα και στην περίπτωση της ύπαρξης πολλών ζεύξεων μεταξύ του ΑΣ3 και του ΑΣ6, ένα λάθος στην διαμόρφωση στο ΑΣ3 θα μπορούσε να αποσυνδέσει το ΑΣ6. Αντίθετα, αν το ΑΣ7 αντιμετωπίσει μια παρόμοια αποτυχία με το ΑΣ3, τότε μπορεί απλά να ανακατευθύνει την κίνησή του στο ΑΣ4.

Τα Αυτόνομα Συστήματα, όταν αυτό είναι δυνατό, προσπαθούν να διαπραγματευτούν το μεριζόμενο κόστος διασύνδεσης, όπως το ΑΣ4 και ΑΣ5. Το ΑΣ4 πρέπει να πληρώσει το ΑΣ1 για την σύνδεση μεταξύ των ΑΣ7 και ΑΣ6 (παρόλο που στην περίπτωση του ΑΣ7 θα χρησιμοποιήσει κατά πάσα πιθανότητα το ΑΣ4 για να επικοινωνήσει με το ΑΣ6 μόνο στην περίπτωση μιας αποτυχίας στην ζεύξη μεταξύ

ΑΣ7 και ΑΣ3). Εντούτοις, έχει την δυνατότητα να επικοινωνήσει με το ΑΣ8 άνευ κόστους (θα μπορούσε να επιλεχθεί και το μονοπάτι ΑΣ4 → ΑΣ1→ ΑΣ2 → ΑΣ5 → ΑΣ8, αλλά δεν προτιμάται λόγω κόστους και επίδοσης).

Στο παρελθόν, οι τελικοί χρήστες χρησιμοποιούσαν μόνο έναν πάροχο για την σύνδεσή τους στο Διαδίκτυο, όπως για παράδειγμα ο πελάτης Π1 στο Σχήμα 1. Αυτή η κατάσταση, με την πάροδο του χρόνου, αλλάζει κυρίως στον τομέα των κινητών επικοινωνιών. Τα «έξυπνα» τηλέφωνα (smartphones) είναι πλέον εξοπλισμένα με διεπαφές 3G/4G και WiFi. Στην σημερινή εποχή, είναι σύνηθες το φαινόμενο πολλές εταιρίες, ακόμα και ιδιώτες, να αποκτούν σύνδεση στο Διαδίκτυο από δύο παρόχους, όπως για παράδειγμα ο πελάτης Π3 στο Σχήμα 1, για την βελτίωση της αντοχής της σύνδεσης τους. Τα Κέντρα Δεδομένων διαθέτουν πλέον πολλαπλές συνδέσεις, για να επιτευχθεί η εξισορρόπηση του φορτίου και η ανθεκτικότητα σε αποτυχίες. Τέλος, μελέτες δείχνουν ότι οι τεχνικές εξισορρόπησης του φορτίου ανά ροή χρησιμοποιούνται ευρέως στο σημερινό Διαδίκτυο [Aug10].



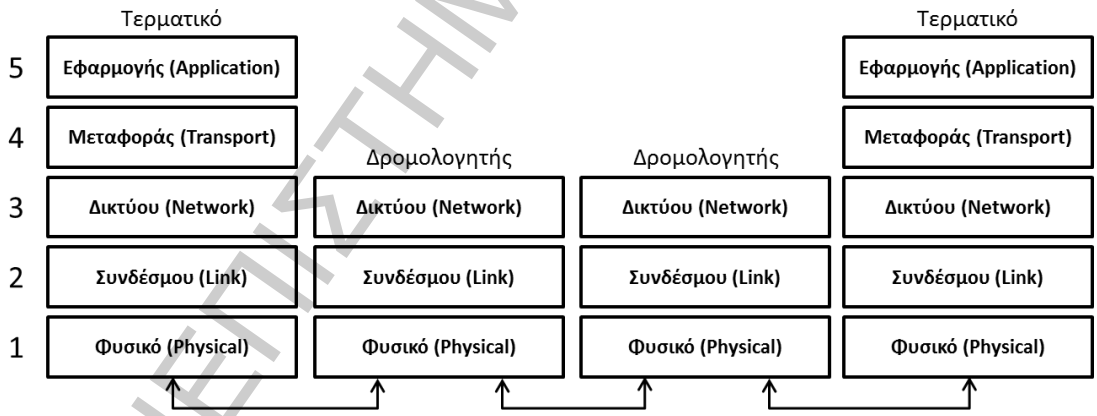
Σχήμα 1: Παράδειγμα διασύνδεσης Αυτόνομων Συστημάτων

Γενικώς, **ένα Αυτόνομο Σύστημα αποκαλείται Σύστημα Πολλαπλών Συνδέσεων (multihomed) όταν έχει την δυνατότητα να παρέχει συνδεσιμότητα στους πελάτες του μέσω περισσοτέρων από μια σύνδεση [deL05].** Χαρακτηριστικό παράδειγμα Συστήματος Πολλαπλών Συνδέσεων είναι ο Π3, όπου διαθέτει 2 συνδέσεις με τα ΑΣ7 και ΑΣ8. Αντιθέτως, ο Π2 *δεν* χαρακτηρίζεται ως Σύστημα Πολλαπλών Συνδέσεων, λόγω της ύπαρξης μιας και μόνο σύνδεσης με το Διαδίκτυο, μέσω του ΑΣ7.

1.2 Multihoming στο IPv4

Το πρωτόκολλο BGP είναι το πρωτόκολλο που μεταφράζει τις επιχειρηματικές σχέσεις σε δικτυακές συνδέσεις. Το Πρωτόκολλο Διαδικτύου (IP) [Pos81a], είναι το πρωτόκολλο που μεταφέρει τα δεδομένα με βάση τις πληροφορίες πρόσβασης που παρέχονται από το πρωτόκολλο BGP. Με άλλα λόγια, το BGP αποτελεί μέρος του επιπέδου ελέγχου του Διαδικτύου, ενώ το Πρωτόκολλο Διαδικτύου (IP) αποτελεί μέρος του επιπέδου λειτουργίας μεταφοράς δεδομένων. Το επίπεδο λειτουργίας μεταφοράς δεδομένων απεικονίζεται στο Σχήμα 2. Τα επίπεδα αναφέρονται με το όνομά τους ή τον αριθμό επιπέδου. Το επίπεδο 1 και 2 σχετίζονται με την πρόσβαση σε μέσα, ασύρματα (WiFi, 3G/4G, κ.ά.) ή ενσύρματα (Ethernet, κ.ά.). Το επίπεδο 3 είναι το επίπεδο δικτύου, όπου παρέχει μη αξιόπιστη παράδοση πακέτων δεδομένων μεταξύ δύο τελικών χρηστών, οπουδήποτε στο Διαδίκτυο. Σήμερα, το Πρωτόκολλο Διαδικτύου (IP) είναι η πιο ευρέως χρησιμοποιούμενη τεχνολογία επιπέδου 3. Το επίπεδο 4 (επίπεδο μεταφοράς) βασίζεται στο επίπεδο 3 για τον εντοπισμό του προορισμού, που προσδιορίζεται από μια διεύθυνση, και την παράδοση των πακέτων. Γενικώς, το επίπεδο χ βασίζεται στις υπηρεσίες που παρέχονται από το επίπεδο $\chi - 1$. Ενώ ο σκοπός του επιπέδου δικτύου είναι ο εντοπισμός του συστήματος προορισμού στο Διαδίκτυο, ο σκοπός του επιπέδου μεταφοράς είναι η δρομολόγηση των δεδομένων σε μια εφαρμογή, που αναφέρεται από έναν αριθμό πόρτας, εντός του συστήματος προορισμού. Παρέχει επίσης την δυνατότητα της αξιόπιστης, και σε ορθή σειρά, παράδοσης των δεδομένων στο επίπεδο εφαρμογής. Για παράδειγμα, το πρωτόκολλο UDP [Pos80] εγγυάται για την ακεραιότητα των μεταδιδόμενων δεδομένων, χρησιμοποιώντας ένα άθροισμα ελέγχου (checksum), αλλά όχι την σειρά ή την αξιόπιστη παράδοση αυτών. Αντιθέτως, το πρωτόκολλο TCP [Pos81b] εγγυάται

για την ακεραιότητα των δεδομένων, χρησιμοποιώντας όπως και στο UDP ένα άθροισμα ελέγχου (checksum), την σειρά παράδοσης, χρησιμοποιώντας αριθμούς ακολουθίας, και την αξιόπιστη παράδοση τους, χρησιμοποιώντας χρονομετρητές και επαναποστέλλοντας τα δεδομένα που δεν έχουν παραδοθεί. Επιπλέον, το πρωτόκολλο TCP έχει την δυνατότητα να προσαρμόζει δυναμικά τον ρυθμό αποστολής στην χωρητικότητα της ζεύξης και τα επίπεδα συμφόρησης, χάρη σε έναν μηχανισμό ελέγχου συμφόρησης. Σήμερα, πάνω από το 95% της συνολικής κίνησης του Διαδικτύου χρησιμοποιεί για το επίπεδο μεταφοράς το πρωτόκολλο TCP ή UDP [LIJM+10]. Τέλος, το επίπεδο εφαρμογής παρέχει συγκεκριμένα πρωτόκολλα που προσαρμόζονται στις ανάγκες των εκάστοτε εφαρμογών, όπως είναι το HTTP για την πλοήγηση στις διαδικτυακές σελίδες, το FTP για την μεταφορά αρχείων, κ.ά.

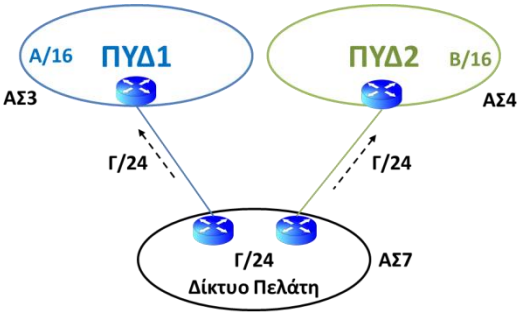


Σχήμα 2: Επίπεδα λειτουργίας τερματικών και δρομολογητών ως προς το μοντέλο αναφοράς TCP/IP

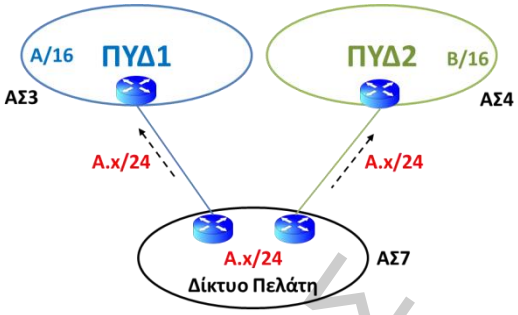
IP Multihoming με μπλοκ διευθύνσεων ανεξαρτήτου παρόχου (Provider Independent addresses): Είναι γνωστό ότι οι διευθύνσεις της τέταρτης έκδοσης του Πρωτοκόλλου Διαδικτύου (IPv4) έχουν μέγεθος 32-bit. Ένα Αυτόνομο Σύστημα

μπορεί να ζητήσει να του ανατεθεί ένα σύνολο (μπλοκ) διευθύνσεων από το Περιφερειακό Μητρώο Διαδικτύου (Regional Internet Registry – RIR), όπως είναι για παράδειγμα το RIPE NCC (Réseaux IP Européens Network Coordination Centre) στην Ευρώπη. Αν το ΑΣ7, στο Σχήμα 1, λάβει το πρόθεμα Γ/24, τότε αυτό θα κοινοποιηθεί στους παρόχους ΑΣ3 και ΑΣ4, μέσω του πρωτοκόλλου BGP. Η διαδικασία αυτή απεικονίζεται στο Σχήμα 3. Η κοινοποίηση τότε διαδίδεται έως ότου καταστεί γνωστό σε όλο το Διαδίκτυο ότι το ΑΣ7 μπορεί να προσπελαστεί μέσω των ΑΣ3 ή ΑΣ4. Σημειώνεται δε ότι μέρος του Διαδικτύου θα γνωρίζει μόνο ένα ΑΣ εξ' αυτών. Για παράδειγμα, το ΑΣ1 θα λάβει μια κοινοποίηση BGP για το πρόθεμα Γ/24 τόσο από το ΑΣ3 όσο και από το ΑΣ4. Ωστόσο, το ΑΣ1 θα εκτελέσει τον αλγόριθμο απόφασης BGP και στη συνέχεια θα κοινοποιήσει στα γειτονικά ΑΣ μόνο ένα εκ των διαθέσιμων μονοπατιών, που θεωρήθηκε ως το βέλτιστο από τον αλγόριθμο.

Το multihoming με μπλοκ διευθύνσεων ανεξαρτήτου παρόχου δεν αποτελεί πλέον κατάλληλη τεχνική για δύο λόγους. Αφενός, ο χώρος διευθύνσεων του IPv4 έχει πλέον εξαντληθεί [NRO10], γι' αυτό τώρα είναι σχεδόν αδύνατο να λάβει κάποιο μπλοκ διευθύνσεων ανεξαρτήτου παρόχου από το Περιφερειακό Μητρώο Διαδικτύου. Αφετέρου, η χρήση των προθεμάτων ανεξαρτήτου παρόχου, σε αντίθεση με το πρόθεμα ομαδοποιημένων παρόχων, δεν επιτρέπει την ομαδοποίηση των προθεμάτων, και ως εκ τούτου συμβάλει στην ανάπτυξη των πινάκων δρομολόγησης του Διαδικτύου [ALD^{+}05].



Σχήμα 3: Multihoming με μπλοκ διευθύνσεων ανεξαρτήτου παρόχου



Σχήμα 4: Multihoming με μπλοκ διευθύνσεων ομαδοποιημένων παρόχων

IP Multihoming με μπλοκ διευθύνσεων ομαδοποιημένων παρόχων (Provider Aggregatable addresses): Σε αντίθεση με τις διευθύνσεις ΑΠ, οι οποίες λαμβάνονται από το ΠΜΔ, οι διευθύνσεις ομαδοποιημένων παρόχων λαμβάνονται από τον πάροχο, όπως απεικονίζεται στο Σχήμα 4. Στο σχήμα αυτό, το ΑΣ7 λαμβάνει ένα μπλοκ διευθύνσεων, Α.x/24, από τον πάροχο ΠΥΔ1 (ΑΣ3), το οποίο είναι υποσύνολο του χώρου διευθύνσεων που έχει στην διάθεσή του ο ΠΥΔ1 (Α/16). Αυτό μετριάζει το πρόβλημα εξάντλησης των διευθύνσεων που αναφέρθηκε προηγουμένως, παρόλο που ο ΠΥΔ1 δεν διαθέτει άπειρες διευθύνσεις και είναι τελικά εξαρτημένο από το ΠΜΔ. Εκτός αυτού, η κατανομή των διευθύνσεων ΟΠ επιτρέπει την καλύτερη ομαδοποίηση των προθεμάτων, που οδηγεί σε μικρότερους πίνακες δρομολόγησης. Ο ΠΥΔ1 χρειάζεται να κοινοποιήσει ένα πρόθεμα (Α/16) αντί για δύο στην περίπτωση των ΑΠ (Α/16 και Γ/24).

Ωστόσο, το multihoming καταργεί αυτό το πλεονέκτημα. Όταν ένα σύστημα είναι multihomed, αυτό δεν επιθυμεί να ζητήσει ένα πρόθεμα IPv4 από κάθε ένα από τους παρόχους του, λόγω της έλλειψης των διευθύνσεων, το οποίο μεταφράζεται σε μεγάλο οικονομικό κόστος. Επίσης, οι τρέχουσες στοίβες δικτύου στους τελικούς χρήστες δεν είναι σχεδιασμένες για να διαχειριστούν αποτελεσματικά τις πολλαπλές

διευθύνσεις IPv4 που έχουν αποδοθεί. Μια κοινή τεχνική IPv4 multihoming, για κόμβους με διευθύνσεις ΟΠ, είναι η κοινοποίηση των προθεμάτων ΟΠ σε κάθε εταίρο, ακριβώς σαν να ήταν μπλοκ διευθύνσεων ΑΠ. Δυστυχώς, αυτή η τεχνική συμβάλλει στην ανάπτυξη των πινάκων δρομολόγησης, επειδή ο ΠΥΔ2 πρέπει να κοινοποιήσει και το πρόθεμα A.x/24. Αυτό μπορεί να συμβεί και στην περίπτωση που το μέγεθος του προθέματος είναι πολύ μεγάλο, μεγαλύτερο από το /24, και να φιλτραριστεί από τους ΑΣ διέλευσης. Σε αυτή τη περίπτωση, ο multihomed κόμβος, χάνει μέρος των πλεονεκτημάτων που θα ανέμενε από το multihoming [ALD$^+$05]. Τέλος, οι πελάτες χρειάζονται να λάβουν νέες διευθύνσεις στην περίπτωση όπου επιθυμούν την αλλαγή του κύριου παρόχου.

1.3 Multihoming στο Επίπεδο Μεταφοράς

Το multihoming σε επίπεδο δικτύου παρέχει δυνατότητα ανάκτησης μετά από κάποια αποτυχία. Ωστόσο, δεν μπορεί να χρησιμοποιηθεί ως ένας τρόπος για να αξιοποιηθούν ταυτόχρονα πολλές ζεύξεις για μια μόνο σύνδεση μεταφοράς. Για να επιτευχθεί η αποτελεσματική εξισορρόπηση της κίνησης σε πολλαπλές ζεύξεις, απαιτείται η τροποποίηση του επιπέδου μεταφοράς. Ένα πλεονέκτημα της τροποποίησης του επιπέδου μεταφοράς αντί του επιπέδου δικτύου είναι ότι το προκύπτον πρωτόκολλο πολυδιαδρομικής μεταφοράς μπορεί να χρησιμοποιηθεί ανεξάρτητα από το πρωτόκολλο που θα χρησιμοποιηθεί στο επίπεδο δικτύου, όπως για παράδειγμα το IPv4 ή το IPv6.

Στο πρόσφατο παρελθόν έχουν επιχειρηθεί πολλές προσπάθειες για την επίτευξη δημιουργίας ενός πολυδιαδρομικού πρωτοκόλλου, πρώτα ως επέκταση του πρωτοκόλλου TCP [MK01, HS02, ROA05, ZLK04]. Παρόλα αυτά, αυτές οι επεκτάσεις ουδέποτε υλοποιήθηκαν ή είχαν εγκατασταθεί ευρέως στο Διαδίκτυο. Το Πρωτόκολλο Ελέγχου Ροής της Μετάδοσης (Stream Control Transmission Protocol – SCTP) [Ste07] είχε σχεδιασθεί με γνώμονα την ύπαρξη πολλαπλών συνδέσεων (multihoming) και την δυνατότητα αυτόματης αλλαγής σε περίπτωση αποτυχίας (failover). Πολλές επεκτάσεις του πρωτοκόλλου SCTP [IAS06, ASL04, LWZ08] επιτρέπουν στο σύστημα να χρησιμοποιεί πολλαπλές διεπαφές ταυτόχρονα. Παρόλο που υλοποιήθηκε σε αρκετά συστήματα [IAS06], εντούτοις το πρωτόκολλο SCTP δεν χρησιμοποιείται ευρέως σήμερα, εξαιρουμένων μερικών εφαρμογών. Τα βασικά μειονεκτήματα του SCTP στο παγκόσμιο Διαδίκτυο είναι αφενός ότι οι προγραμματιστές εφαρμογών πρέπει να αλλάξουν την εφαρμογή τους, ούτως ώστε να χρησιμοποιεί ως πρωτόκολλο μεταφοράς το SCTP, και αφετέρου ενδιάμεσοι κόμβοι (middleboxes) διαφόρων τύπων, όπως για παράδειγμα το NAT ή το Firewall, δεν υποστηρίζουν το συγκεκριμένο πρωτόκολλο, με αποτέλεσμα να μπλοκάρονται τα πακέτα που το χρησιμοποιούν.

Κατά τη διάρκεια των δύο τελευταίων ετών, η ομάδα εργασίας MPTCP του οργανισμού IETF έχει αναπτύξει επεκτάσεις πολυδιαδρομικότητας στο TCP [FRHB12], οι οποίες επιτρέπουν σε συστήματα να χρησιμοποιούν αρκετά μονοπάτια, πιθανώς μέσω πολλαπλών διεπαφών, για την μεταφορά πακέτων που ανήκουν σε μια μόνο σύνδεση. Αυτή είναι ίσως και η πιο φιλόδοξη επέκταση στο TCP που θα τυποποιηθεί από τον IETF.

Το Multipath TCP [FRHB12] διαφέρει ως προς της υπάρχουσες επεκτάσεις του TCP, όπως για παράδειγμα στα μεγάλα παράθυρα αποστολής/λήψης, στις χρονοσφραγίδες

και στις επιλεκτικές επιβεβαιώσεις (Selective Acknowledgment). Αυτές οι παλαιότερες επεκτάσεις ορίζουν νέα TCP Options, τα οποία αλλάζουν ελαφρώς την αντίδραση των συστημάτων όταν τα λαμβάνουν. Το Multipath TCP επιτρέπει σε ένα ζεύγος συστημάτων να χρησιμοποιούν διάφορα μονοπάτια για την ανταλλαγή των πακέτων που μεταφέρουν τα δεδομένα μιας και μόνο σύνδεσης.

Όταν η εφαρμογή ανοίγει ένα νέο TCP socket σε μια στοίβα πολυδιαδρομικών πρωτοκόλλων, η στοίβα στην πραγματικότητα ανακαλύπτει τον αριθμό των μονοπατιών που είναι διαθέσιμα για την επικοινωνία με τον προορισμό, και ανοίγει όσες TCP υποροές υπαγορεύει η εσωτερική ευρετική μέθοδος (αλγόριθμος), μέχρι το μέγιστο αριθμό των γνωστών μονοπατιών. Η λεπτομερής διαδικασία εγκατάστασης μιας σύνδεσης για το Multipath TCP περιγράφεται στο κεφάλαιο 2.3. Τα δεδομένα που παράγονται από τον πελάτη και τον εξυπηρετητή μπορούν να αποσταλούν σε οποιαδήποτε από τις υποροές που συνθέτουν μια σύνδεση Multipath TCP και στην περίπτωση όπου μια υποροή αποτύχει, τότε τα δεδομένα μπορούν να αποσταλούν εκ νέου από κάποια άλλη υποροή. Για την επίτευξη αυτού, το Multipath TCP βασίζεται σε δύο αρχές:

- Κάθε υποροή είναι ισοδύναμη με μια κανονική σύνδεση TCP με το δικό της χώρο ακολουθίας αριθμών (μεγέθους 32bit). Αυτό είναι σημαντικό για να επιτραπεί στο πρωτόκολλο να διασχίσει πολύπλοκους ενδιάμεσους κόμβους (middleboxes), όπως είναι για παράδειγμα οι διαμεσολαβητές ή οι κανονικοποιητές κίνησης.
- Το Multipath TCP διατηρεί έναν χώρο ακολουθίας αριθμών μεγέθους 64bit. Όταν ένα σύστημα αποστέλλει ένα τμήμα TCP μέσω μιας υποροής, αυτό υποδεικνύει μέσα στο τμήμα, χρησιμοποιώντας το TCP Option "Σήμα Ακολουθίας Δεδομένων" (Data Sequence Signal – DSS [FRHB12]), την

συσχέτιση μεταξύ του 64bit αριθμού ακολουθίας δεδομένων και του 32bit αριθμού ακολουθίας που χρησιμοποιείται από την υποροή. Χάρη σε αυτή τη συσχέτιση, ο δέκτης μπορεί να ανακατατάξει τα ληφθέντα δεδομένα στη σωστή σειρά, που πιθανότατα έχουν ληφθεί εκτός σειράς από διάφορες υποροές.

Στο Multipath TCP, η επιβεβαίωση για την λήψη ενός τμήματος πραγματοποιείται σε δύο επίπεδα:

- Το TCP επιβεβαιώνει συσσωρευτικά ή επιλεκτικά για την επιτυχή λήψη των τμημάτων σε κάθε υποροή.
- Η επιβεβαίωση σε επίπεδο σύνδεσης επιστρέφεται από τον παραλήπτη για την παροχή συσσωρευτικής επιβεβαίωσης σε επίπεδο ακολουθίας δεδομένων.

Το ίδιο TCP Option "Σήμα Ακολουθίας Δεδομένων" (DSS) χρησιμοποιείται για την πληροφόρηση του εταίρου σχετικά με τον αριθμό ακολουθίας επιπέδου σύνδεσης και την επιβεβαίωση επιπέδου σύνδεσης. Όταν ένα τμήμα χάνεται, ο δέκτης ανιχνεύει το κενό στο ληφθέντα αριθμό ακολουθίας (32bit) και ενεργοποιείται ο (υφιστάμενος) μηχανισμός επαναποστολής του πρωτοκόλλου TCP. Όταν μια υποροή αποτύχει, τότε το Multipath TCP ανιχνεύει την αποτυχία και επανεκπέμπει τα μη-επιβεβαιωμένα δεδομένα μέσω μιας άλλης υποροής που είναι ακόμα ενεργή.

Μια ακόμη σημαντική διαφορά μεταξύ του Multipath TCP και του κανονικού TCP είναι το σύστημα ελέγχου συμφόρησης. Το Multipath TCP δεν μπορεί να χρησιμοποιήσει το σύστημα ελέγχου του TCP χωρίς να αδικεί τις κανονικές TCP ροές. Για παράδειγμα, δύο συστήματα μοιράζονται μια συμφορημένη σύνδεση. Αν και τα συστήματα χρησιμοποιούν το κανονικό TCP και εγκαταστήσουν μια νέα σύνδεση TCP, τότε θα πρέπει να επιτύχουν σχεδόν την ίδια απόδοση. Αν ένα

σύστημα εγκαταστήσει πολλαπλές υποροές για μια σύνδεση Multipath TCP, οι οποίες διέρχονται όλες από την συμφορημένη σύνδεση, τότε δεν θα πρέπει να είναι σε θέση να χρησιμοποιήσει περισσότερο από το μερίδιο της στη σύνδεση. Αυτό επιτυγχάνεται με τον συζευγμένο σύστημα ελέγχου συμφόρησης το οποίο αναλύεται διεξοδικά στα [RHW11, WRGH11]. Ο έλεγχος συμφόρησης του κανονικού TCP [APB09] αυξομειώνει το παράθυρο συμφόρησης και το κατώφλι του αλγορίθμου αργής εκκίνησης (slow-start) κατά την παραλαβή των μηνυμάτων επιβεβαίωσης και της ανίχνευσης των απωλειών. Το συζευγμένο σύστημα ελέγχου συμφόρησης βασίζεται επίσης σε ένα παράθυρο συμφόρησης, αλλά ενημερώνεται σύμφωνα με την ακόλουθη αρχή [RHW11]:

- Για κάθε μη-διπλή επιβεβαίωση στην υποροή *i*, αυξάνεται το μέγεθος του παραθύρου συμφόρησης της υποροής *i* κατά:

$$\min\left(\frac{\alpha * bytes\_acked * mss_i}{cwnd_{tot}}, \frac{bytes\_acked * mss_i}{cwnd_i}\right),$$

όπου $cwnd_{tot}$ είναι το συνολικό παράθυρο συμφόρησης όλων των υποροών και

$$\alpha = cwnd_{tot} \frac{max_i\left(\frac{cwnd_i}{RTT_i^2}\right)}{\left(\sum_i \frac{cwnd_i}{RTT_i}\right)^2}$$

- Κατά τον εντοπισμό της απώλειας στην υποροή *i*, το παράθυρο συμφόρησης της υποροής μειώνεται κατά $cwnd_i/2$

Η επίτευξη της δικαιοσύνης μεταξύ ροών Multipath TCP και κανονικού TCP πραγματοποιείται στον παραπάνω αλγόριθμο με τον περιορισμό των παραθύρων με δύο τρόπους:

- Οι αυξήσεις στο παράθυρο συμφόρησης στο Multipath TCP έχουν το ίδιο ανώτατο όριο με τις αντίστοιχες αυξήσεις στο TCP. Αυτός ο περιορισμός εξασφαλίζει ότι το Multipath TCP δεν θα λάβει περισσότερο από το διαθέσιμο εύρος ζώνης σε σύγκριση με τις κανονικές ροές TCP, σε οποιαδήποτε υποροή της.
- Η παράμετρος α ελέγχει τον ρυθμό αύξησης. Η σύνθεση της παραμέτρου προέρχεται από την επίλυση της εξίσωσης ισορροπίας, υπό τον περιορισμό ότι κάθε συνδυασμός μονοπατιών δεν μπορεί να έχει μεγαλύτερη χωρητικότητα από μια κανονική ροή TCP, με την χρήση των καλύτερων, από τα προσφερόμενα, μονοπατιών. Αυτό αποτρέπει ένα σύνολο πολλαπλών υποροών, που μοιράζονται μια συμφορημένη σύνδεση, από την λήψη μεγαλύτερης χωρητικότητας από τις ανταγωνιστικές ροές TCP [WRGH11].

ΚΕΦΑΛΑΙΟ 2

ΑΝΑΛΥΣΗ ΤΟΥ MULTIPATH TCP

2.1 Το πρωτόκολλο TCP

Το TCP (Transmission Control Protocol – Πρωτόκολλο Ελέγχου Μεταφοράς), που έγινε πρότυπο το 1981 [Pos81b], χρησιμοποιείται στη συντριπτική πλειοψηφία των επικοινωνιών στο Διαδίκτυο. Μαζί με το πρωτόκολλο UDP (User Datagram Protocol), μεταφέρουν το 95% της κίνησης του δικτύου στο Διαδίκτυο [LIJM+10].

Το πρωτόκολλο TCP παρέχει αξιόπιστη αποστολή και λήψη δεδομένων, μεταφέροντας αυτά χωρίς λάθη μεταξύ του επιπέδου δικτύου και του επιπέδου εφαρμογής και παραδίδοντας τα δεδομένα στο επίπεδο εφαρμογής στη σωστή σειρά.

Ο αποστολέας γνωρίζει ότι μια σειρά από bytes έχουν φθάσει σωστά στον προορισμό, χάρη στα μηνύματα επιβεβαίωσης `ACK` (`ACKnowledgment`). Στα bytes δεδομένων αποδίδονται κάποιοι αριθμοί ακολουθίας που αναφέρονται στα μηνύματα επιβεβαίωσης και υποδεικνύουν το πλήθος των ορθών δεδομένων που έχουν ληφθεί. Για παράδειγμα, μια επιβεβαίωση για το *1000ο* byte σημαίνει ότι τα bytes με αύξων αριθμό από *0* μέχρι *999* έχουν ληφθεί επιτυχώς.

Το TCP εκτιμά τον χρόνο μετ' επιστροφής (RTT – Round Trip Time) ως τον χρόνο που έχει παρέλθει μεταξύ της μετάδοσης ενός τμήματος και την λήψη της αντίστοιχης επιβεβαίωσης. Το RTT χρησιμοποιείται για να διαμορφώσει ένα χρονόμετρο που θα ενεργοποιήσει μια αυτόματη διαδικασία επαναποστολής των δεδομένων, όταν αυτό λήξει. Όλα αυτά εκτελούνται χωρίς η εφαρμογή να έχει επίγνωση, η οποία απλώς τροφοδοτεί ένα socket με δεδομένα και είναι βέβαιη για την αξιόπιστη παράδοση, εφόσον το δικτυακό μονοπάτι που χρησιμοποιείται για την επικοινωνία είναι προσβάσιμο.

Bits	**0 - 3**	**4 -9**	**10 – 15**	**16 - 31**
0	**Source Port** Θύρα Προέλευσης			**Destination Port** Θύρα Προορισμού
32	**Sequence Number** Αριθμός ακολουθίας			
64	**Acknowledgement Number** Αριθμός επιβεβαίωσης			
96	**Data Offset**	**Reserved**	**Flags** Σημαίες	**Window** Παράθυρο
128	**Checksum** Άθροισμα ελέγχου			**Urgent Pointer** Επείγοντα δεδομένα
160	**Options** Επιλογές			
160/192+	**Data** Δεδομένα			

Πίνακας 1: Επικεφαλίδα πρωτοκόλλου TCP

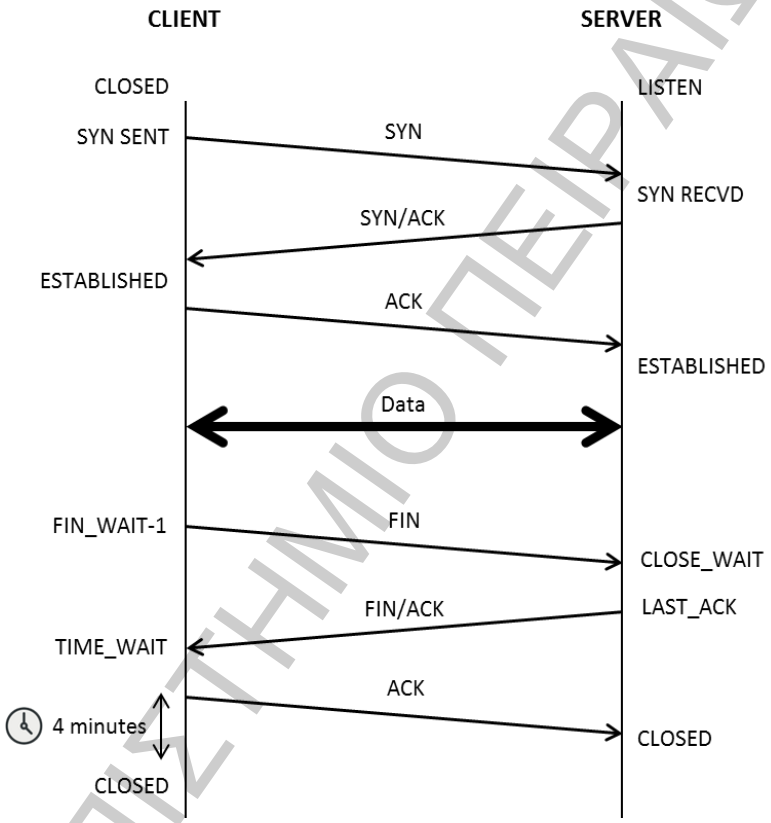
Ο όρος «ροή bytes» (bytestream) σημαίνει ότι το πρωτόκολλο TCP επιτρέπει στο επίπεδο εφαρμογής να ανταλλάξει μια ροή δεδομένων με τον ομόλογό του, η οποία

απαιτεί κατάτμηση, επειδή το υποκείμενο επίπεδο δικτύου μπορεί να χειριστεί μόνο τμήματα δεδομένων σε μορφή πακέτων. Το πρωτόκολλο TCP έχει την δυνατότητα να διαπραγματευτεί για το Μέγιστο Μέγεθος Τμήματος (MSS – Maximum Segment Size) στο ξεκίνημα μιας σύνδεσης, και στη συνέχεια προσπαθεί να καλύψει όλα τμήματα με το μέγιστο μέγεθος, προκειμένου να επωφεληθούν από την καλύτερη δυνατή αναλογία ελέγχου/δεδομένων σε κάθε τμήμα, χάρη στον αλγόριθμο Nagle [Nag84].

Στο RFC 793 [Pos81b] δίδεται μεγάλη έμφαση στις φάσεις εγκατάστασης και τερματισμού μιας σύνδεσης TCP και καθορίζεται μια πλήρης μηχανή καταστάσεων. Στο Σχήμα 5 απεικονίζεται ένα απλό παράδειγμα μιας σύνδεσης TCP, αναγράφοντας τα ονόματα των καταστάσεων και των μηνυμάτων ανταλλαγής, όπως αυτά περιγράφονται στο RFC 793 [Pos81b]

Ένας TCP διακομιστής αναμένει για εισερχόμενες αιτήσεις σύνδεσης. Μέχρι να παραληφθεί ένα αίτημα, ο διακομιστής βρίσκεται σε κατάσταση `LISTEN`. Η κατάσταση `CLOSED` από την πλευρά του πελάτη αναφέρεται στην απουσία socket. Για την εγκαθίδρυση μιας σύνδεσης, απαιτείται η ανταλλαγή τριών μηνυμάτων, διαδικασία που είναι ευρέως γνωστή και ως «Τριμερής Χειραψία» (Three-Way Handshaking). Το πρώτο μέρος για την επίτευξη μιας σύνδεσης είναι το μήνυμα συγχρονισμού `SYN` (`SYNchronize`). Ο διακομιστής λαμβάνει το μήνυμα συγχρονισμού και αυτός με την σειρά του αποστέλλει ένα μήνυμα επιβεβαίωσης, ονόματι `SYN-ACK` (`SYNchronize-ACKnowledgement`). Σε περίπτωση όπου το μήνυμα συγχρονισμού δεν ληφθεί από τον διακομιστή, τότε αποστέλλεται εκ νέου από τον πελάτη. Τέλος, ο πελάτης αποστέλλει ένα μήνυμα επιβεβαίωσης προς τον διακομιστή, εγκαθιδρύοντας με αυτό το τρόπο μια σύνδεση. Ο διακομιστής δεν

εισέρχεται σε κατάσταση ESTABLISHED, έως ότου επιβεβαιωθεί το δικό του μήνυμα συγχρονισμού SYN-ACK. Εισέρχοντας στην κατάσταση ESTABLISHED πριν την λήψη της επιβεβαίωσης ACK (ACKnowledgement) θα οδηγούσε δυνητικά σε αποστολή δεδομένων που ο δέκτης δεν θα ήταν σε θέση να εντοπίσει στην λίστα όπου καθορίζεται η σειρά.



Σχήμα 5: Παράδειγμα ανταλλαγής μηνυμάτων του πρωτοκόλλου TCP

Μια παρόμοια διαδικασία ακολουθείται και στον τερματισμό μιας σύνδεσης. Όταν η εφαρμογή καλέσει την μέθοδο συστήματος close() για το TCP Socket, τότε αποστέλλεται ένα μήνυμα τερματισμού FIN (FINal). Η λήψη του μηνύματος τερματισμού FIN οδηγεί στην κλήση της μεθόδου close() του συστήματος στην εφαρμογή από την πλευρά του διακομιστή, το οποίο με την σειρά του αποστέλλει ένα

μήνυμα επιβεβαίωσης FIN-ACK (FINal-ACKnowledgment). Τέλος, ο πελάτης, μετά την λήψη του FIN-ACK, αποστέλλει ένα μήνυμα ACK, που σηματοδοτεί και το τέλος της σύνδεσης για τον διακομιστή. Ο πελάτης δεν δύναται να κλείσει το socket απευθείας με την έκδοση της επιβεβαίωσης ACK, επειδή ενδέχεται ο διακομιστής να αποστείλει εκ νέου μήνυμα FIN-ACK, γνωστοποιώντας προς τον πελάτη για την μη λήψη του μηνύματος ACK. Για τον λόγο αυτό, το κλείσιμο του socket από την πλευρά του πελάτη πραγματοποιείται σε μεταγενέστερο χρόνο, συνήθως μετά το πέρας περιόδου 4 λεπτών από την στιγμή της αποστολής του μηνύματος επιβεβαίωσης ACK, όπως προτείνεται στο [Pos81b].

Στο [Pos81b] καθορίζονται κάποιες σημαντικές μεταβλητές κατάστασης. Μερικές από αυτές, που θα αναφερθούν επίσης και στην περίπτωση του Multipath TCP, είναι οι εξής:

- *RCV.NXT*: Επόμενος αναμενόμενος αριθμός σειράς. Αυτό σημαίνει ότι όλα τα bytes μέχρι και το *RCV.NXT − 1* έχουν ληφθεί επιτυχώς.
- *RCV.WND*: Παράθυρο λήψης. Το πρωτόκολλο TCP εξασφαλίζει την μετάδοση των δεδομένων στην σωστή σειρά, χάρη στον αριθμό ακολουθίας. Στην περίπτωση όπου πακέτα έχουν φθάσει εκτός σειράς, τότε αναδιατάσσονται στην προσωρινή μνήμη (buffer) του δέκτη. Ιδίως, αν ένα πακέτο έχει χαθεί, ένας δέκτης TCP πρέπει να έχει την δυνατότητα να αποθηκεύει όλα τα δεδομένα που ακολουθούν μετά το χαμένο πακέτο, έως ότου αναμεταδοθεί. Αυτό οδηγεί στην γρήγορη συμπλήρωση της προσωρινής μνήμης (buffer) του δέκτη. Μια ακόμα περίπτωση όπου απαιτείται η διαδικασία προσωρινής αποθήκευσης (buffering) είναι όταν μια εφαρμογή είναι πολύ αργή στην ανάγνωση των εισερχομένων δεδομένων. Σε αυτές τις

περιπτώσεις όπου χρειάζεται η διαδικασία προσωρινής αποθήκευσης, είναι σημαντικό να αποφευχθεί η παροχή δεδομένων από τον αποστολέα που δεν μπορούν να αποθηκευτούν. Το παράθυρο λήψης αποτελεί ένδειξη, που συμπεριλαμβάνεται σε κάθε πακέτο, του τρέχοντος διαθέσιμου buffer. Το ακριβές παράθυρο ορίζεται ως:

[RCV.NXT, RCV.NXT + RCV.WND – 1]

- *SND.NXT*: Το επόμενο byte προς αποστολή. Ο αύξων αριθμός αυξάνεται κατά ένα μετά από κάθε νέα αποστολή byte. Αν, για παράδειγμα, ένα πακέτο μεγέθους *1500* bytes πρέπει να αποσταλεί, τότε δίδεται αριθμός ακολουθίας *SND.NXT* και στη συνέχεια ο *SND.NXT* αυξάνεται κατά *1500*.
- *SND.UNA*: Το πρώτο μη-επιβεβαιωμένο byte. Η μεταβλητή αυτή αποθηκεύει το υψηλότερο αθροιστικό *ACK* που έχει ληφθεί την δεδομένη στιγμή. Όλα τα πακέτα με αύξων αριθμό μικρότερο της τιμής της μεταβλητής *SND.UNA* μπορούν απορριφθούν από την προσωρινή μνήμη αποστολής, λόγω της ορθής λήψης από τον δέκτη.
- *SND.WND*: Παράθυρο αποστολής. Η μεταβλητή αυτή είναι η αντίστοιχη μεταβλητή *RCV.WND* από την πλευρά του αποστολέα. Το επιτρεπόμενο παράθυρο αποστολής νέων τμημάτων είναι:

[SND.UNA, SND.UNA + SND.WND – 1]

Έλεγχος συμφόρησης: Έχει αναφερθεί προηγουμένως η περίπτωση μιας ροής TCP που περιορίζεται από την εφαρμογή λήψης. Σε αυτή τη περίπτωση, ο δέκτης διαφημίζει ένα μικρότερο σε μέγεθος παράθυρο λήψης, ώστε να αναγκάσει τον αποστολέα να μην αποστέλλει πιο γρήγορα από τον ρυθμό που λαμβάνει και διαβάζει τα δεδομένα. Είναι πιθανό μια ροή TCP να περιορίζεται από το δίκτυο, γεγονός που

είναι και το πιο σύνηθες. Μια ροή είναι περιορισμένη από το δίκτυο όταν χρησιμοποιεί όλη τη διαθέσιμη χωρητικότητα της ζεύξης που παρουσιάζει συμφόρηση. Σε αυτή τη περίπτωση, οι συμφορημένοι δρομολογητές απορρίπτουν πακέτα, γεγονός που χρήζει επαναποστολής αυτών, όπως έχει εξηγηθεί παραπάνω. Ωστόσο, αν ο ρυθμός αποστολής δεν ελέγχεται, τότε παρουσιάζεται μια πολλαπλασιαστική επενέργεια της συμφόρησης: οι απώλειες επιφέρουν επαναποστολές, σε συνδυασμό με τα νέα δεδομένα που αποστέλλονται παράλληλα στον ίδιο ρυθμό. Ως εκ τούτου, ο συνολικός ρυθμός μετάδοσης αυξάνεται, παρόλο που η απόρριψη κανονικά σημαίνει ότι ο αποστολέας εκπέμπει ήδη πάρα πολύ γρήγορα.

Η λύση για τον έλεγχο συμφόρησης είναι η διατήρηση ενός ξεχωριστού παραθύρου που ονομάζεται παράθυρο συμφόρησης (*cwnd*) [Ste97]. Ο αποστολέας έχει την δυνατότητα να αποστείλει ένα νέο τμήμα δεδομένων μόνο όταν χωράει στο παράθυρο αποστολής αλλά και στο παράθυρο συμφόρησης. Ο έλεγχος συμφόρησης πάντα συμβαίνει σε δύο στάδια. Το πρώτο στάδιο, που ονομάζεται αργή εκκίνηση (*slow start*), χρησιμοποιείται όταν τα επίπεδα συμφόρησης είναι άγνωστα και απαιτείται μια πρώτη διερεύνηση. Ένας πιο ακριβής όρος για αυτό το στάδιο θα μπορούσε να είναι η γρήγορη εκκίνηση, καθώς ο ρυθμός αποστολής αυξάνεται εκθετικά μέχρι την πρώτη απώλεια. Σε αυτή τη φάση, μια ακόμη μεταβλητή, το κατώφλι αργής εκκίνησης, ορίζεται στο μισό μέγεθος του παραθύρου συμφόρησης ($ssthresh = cwnd/2$). Η μεταβλητή *ssthresh* καθορίζει πότε ο αποστολέας πρέπει να είναι πιο συντηρητικός στην αύξηση του παραθύρου συμφόρησης. Αυτό είναι και το δεύτερο στάδιο, που καλείται αποφυγή συμφόρησης (*congestion avoidance*). Στην πρώτη υλοποίηση του αλγορίθμου αποφυγής συμφόρησης (BSD4.3, Tahoe), η αύξηση ήταν γραμμική, μέχρι την πρώτη απώλεια. Ωστόσο, υπάρχουν κι άλλοι αλγόριθμοι

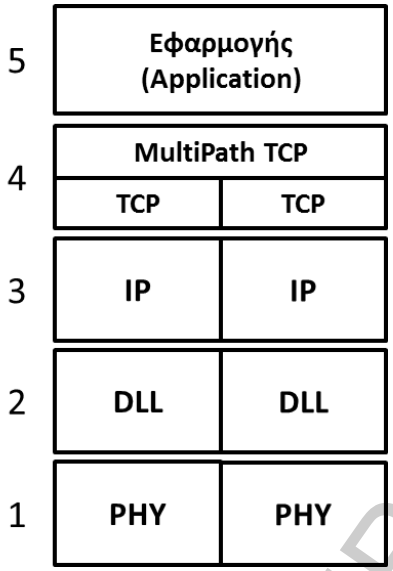
αποφυγής συμφόρησης στις σύγχρονες υλοποιήσεις του πρωτοκόλλου TCP, όπως είναι το TCP Vegas [BOP94], CUBIC [HRX08] ή Illinois [LBS08].

2.2 Πρωτόκολλα πολυδιαδρομικότητας

Το πρωτόκολλο MPTCP είναι η πιο πρόσφατη αλλά δεν είναι η πρώτη προσπάθεια ταυτόχρονης χρήσης πολλαπλών διαδρομών στο επίπεδο μεταφοράς. Προηγούμενες προσπάθειες μπορούν να ταξινομηθούν με βάση τις κατευθύνσεις που λαμβάνονται από τους συγγραφείς. Μια προφανής κατεύθυνση είναι η επιλογή πρωτοκόλλου μεταφοράς: TCP ή SCTP (Πρωτόκολλο Ελέγχου Ροής της Μετάδοσης). Παρατηρείται ότι σε πολλές πτυχές των συμπερασμάτων είναι παρόμοιες, επειδή τα κύρια προβλήματα που αντιμετωπίστηκαν κατά το σχεδιασμό ενός πολυδιαδρομικού πρωτοκόλλου δεν σχετίζονται με τις ιδιαιτερότητες του πρωτοκόλλου SCTP ή του TCP. Αμφότερα τα πρωτόκολλα πρέπει να επιλύσουν τα προβλήματα αναδιάταξης όταν τα δεδομένα έχουν κατανεμηθεί σε πολλαπλές διεπαφές και να προσαρμόσουν την κατανομή της προσωρινής τους μνήμης. Προσεγγίσεις που βασίζονται στην πολυδιαδρομικότητα του πρωτοκόλλου SCTP δικαιολογούν την επιλογή του πρωτοκόλλου στην ικανότητα του SCTP να ορίζει πολλαπλές ροές, οι οποίες μπορούν ευκολότερα να μετατραπούν σε ταυτόχρονες υποροές (subflows). Παρόλα αυτά, εκείνοι που υιοθετούν το πρωτόκολλο TCP [MK01, HS02, ROA05, ZLK04] έχουν δείξει ότι είναι εξίσου εφικτή η μετατροπή του TCP σε ένα πολυδιαδρομικό πρωτόκολλο. Ο λόγος είναι ότι, ενώ το πρωτόκολλο SCTP παρέχει μια διεπαφή socket για τον έλεγχο των υποροών, μια τέτοια διασύνδεση δεν απαιτείται όταν μόνο μια ροή παρουσιάζεται στην εφαρμογή και η διανομή των δεδομένων σε όλες τις

υποροές πραγματοποιείται εσωτερικά. Τελικά επιλέχθηκε το πρωτόκολλο TCP για την υλοποίηση δυνατοτήτων πολυδιαδρομικότητας, επειδή, σε αντίθεση με το SCTP, είναι ευρέως διαδεδομένο στο σημερινό Διαδίκτυο.

Μια ακόμα σημαντική κατεύθυνση είναι η επιλογή του διαστήματος ακολουθίας. Ορισμένες προτάσεις χρησιμοποιούν ένα ενιαίο διάστημα αριθμών ακολουθίας [MK01, ROA05, IAS06]. Αυτή η επιλογή μπορεί να οδηγήσει σε μια σημαντική αναδιάταξη των αριθμών ακολουθίας στο δέκτη. Δεδομένου ότι η αναδιάταξη θεωρείται συνήθως ως μια ένδειξη αποτυχίας, απαιτούνται νέες μέθοδοι ανίχνευσης απώλειας για να γίνει διάκριση μεταξύ μιας κανονικής αναδιάταξης λόγω πολυδιαδρομικότητας και των αποτυχιών. Για την απαλλαγή αυτού του προβλήματος, το πρωτόκολλο MPTCP ορίζει έναν διπλό διάστημα αριθμών ακολουθίας, όπου το ένα διάστημα χρησιμοποιείται αποκλειστικά από την υποροή και προσδιορίζει τα bytes μέσα σε αυτή. Στο [FRHB12], ορίζεται ένας Αριθμός Ακολουθίας Δεδομένων (Data Sequence Number – DNS), ο οποίος φροντίζει για την αναδιάταξη στο συνολικό επίπεδο σύνδεσης, εντούτοις αυτοί οι αριθμοί δεν έχουν καμία επίπτωση στις αποφάσεις επαναποστολής.



Σχήμα 6: Στοίβα πρωτοκόλλων κατά το πρότυπο TCP/IP

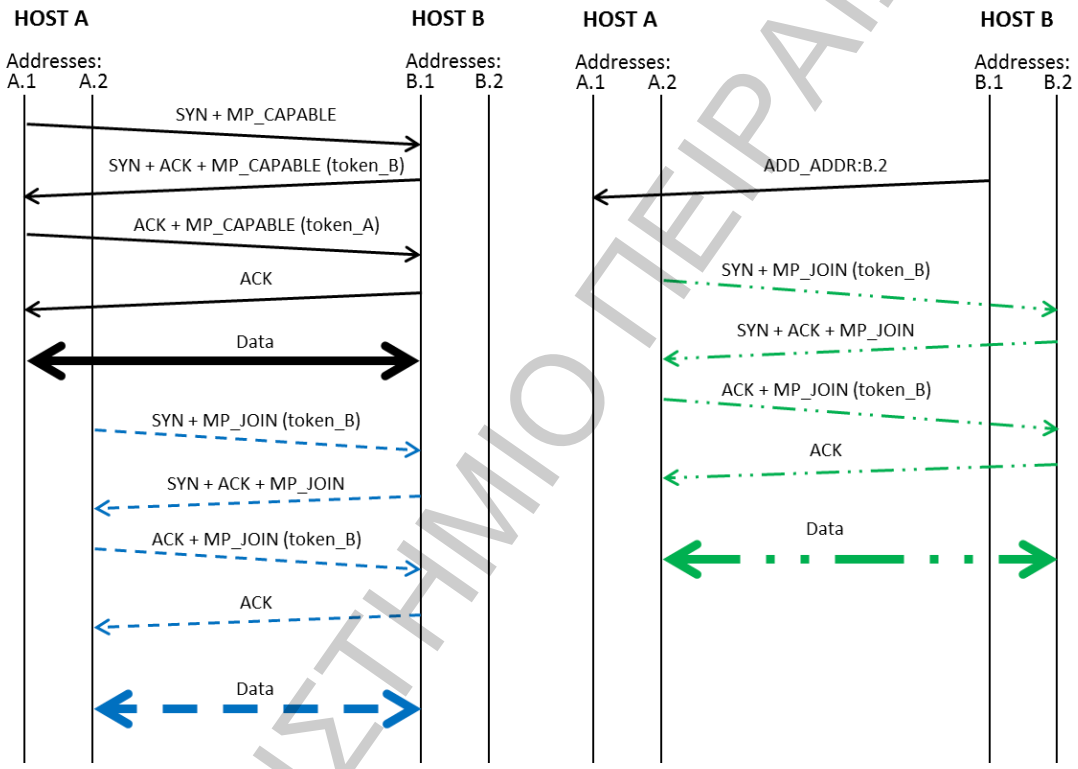
Μια τρίτη επιλογή σχεδιασμού είναι ο τρόπος για την αντιμετώπιση των κοινών σημείων συμφόρησης. Υπάρχει ένα πρόβλημα δικαιοσύνης, όταν πολλές πολυδιαδρομικές ροές μοιράζονται μια συμφόρηση και ενστερνίζονται τον έλεγχο συμφόρησης του TCP σε κάθε υποροή, επειδή θα λάβουν υψηλότερη αναλογία της διαθέσιμης χωρητικότητας σε σύγκριση με τις κανονικές ροές TCP. Στο [ZLK04] επιλύθηκε το πρόβλημα προσπαθώντας να αποφύγουν την εγκαθίδρυση πολλών υποροών κατά μήκος του ίδιου κόμβου συμφόρησης, χάρη σε ένα εξωτερικό εργαλείο. Άλλες προσεγγίσεις αγνοούν απλώς το πρόβλημα. Στο MPTCP, ο αλγόριθμος έλεγχου συμφόρησης συνδέεται σε όλες τις υποροές, ώστε να διασφαλισθεί η δίκαιη κατανομή της ζεύξης χωρίς να απαιτείται η ανίχνευση των κοινών σημείων συμφόρησης [RHW11, WRGH11]. Μια λεπτομερής επισκόπηση των προσεγγίσεων των πολυδιαδρομικών μεταδόσεων μπορεί να βρεθεί στο [Ong09].

2.3 Εκκίνηση νέας συνόδου στο MultiPath TCP

Ένας από τους βασικούς στόχους του σχεδιασμού του πρωτοκόλλου MPTCP είναι να μην γίνεται αντιληπτό τόσο από το επίπεδο εφαρμογής όσο και από το επίπεδο δικτύου. Η εφαρμογή ανοίγει ένα νέο TCP socket, το οποίο αρχικά εκκινεί μια κανονική TCP ροή. Μέχρι αυτό το σημείο δεν υπάρχει καμία σημαντική διαφορά μεταξύ του TCP και του MPTCP. Παρόλα αυτά, αν και οι δύο τελικοί κόμβοι υποστηρίζουν το πρωτόκολλο MPTCP, τότε μπορούν να δημιουργηθούν επιπλέον υποροές. Τα εξερχόμενα δεδομένα προγραμματίζονται σύμφωνα με την πολιτική της εκάστοτε υλοποίησης. Τα εισερχόμενα δεδομένα από όλες τις υποροές αναδιατάσσονται έτσι ώστε να διατηρηθεί η σωστή σειρά.

Το Σχήμα 7 απεικονίζει σε απλουστευμένη μορφή το πρωτόκολλο MPTCP για την περιγραφή της κύριας ιδέας. Η πλήρης περιγραφή του πρωτοκόλλου θα πραγματοποιηθεί στα επόμενα κεφάλαια. Έστω ότι το Σύστημα Α επιθυμεί να επικοινωνήσει με το Σύστημα Β. Από το DNS, μαθαίνει ότι το Σύστημα Β μπορεί να προσπελαστεί μέσω της διεύθυνσης Β.1. Επειδή το Multipath TCP πρέπει να μην γίνεται αντιληπτό από το επίπεδο δικτύου, κάθε νέα υποροή TCP, συμπεριλαμβανομένης και της πρώτης, πρέπει να εγκατασταθεί χρησιμοποιώντας την «Τριμερή Χειραψία» (Three-Way Handshaking). Τα μηνύματα είναι εμπλουτισμένα με ειδικά TCP Options για το Multipath TCP, τα οποία απαιτούνται να ληφθούν υπόψη μόνο από το σύστημα προορισμού και όχι από το δίκτυο. Χάρη στο TCP Option `MP_CAPABLE` που εισάγεται στα μηνύματα `SYN` και `SYN+ACK`, τόσο το Σύστημα Α όσο και το Σύστημα Β μπορούν να ειδοποιηθούν αν η άλλη πλευρά υποστηρίζει το Multipath TCP . Το TCP Option MP_CAPABLE είναι επίσης παρών στο τρίτο μήνυμα της τριμερής χειραψίας (ACK) για να επιτραπεί η αναβολή εκ

μέρους του εξυπηρετητή της εγκατάστασης μιας σύνδεσης Multipath TCP μέχρι την ολοκλήρωση της τριμερής χειραψίας. Αν ένα εκ των δύο τελικών συστημάτων δεν υποστηρίζει το Multipath TCP, τότε το πρωτόκολλο συμπεριφέρεται ακριβώς όπως το TCP.



Σχήμα 7: Αρχική ανταλλαγή μηνυμάτων του πρωτοκόλλου MPTCP και εγκαθίδρυση υποροών

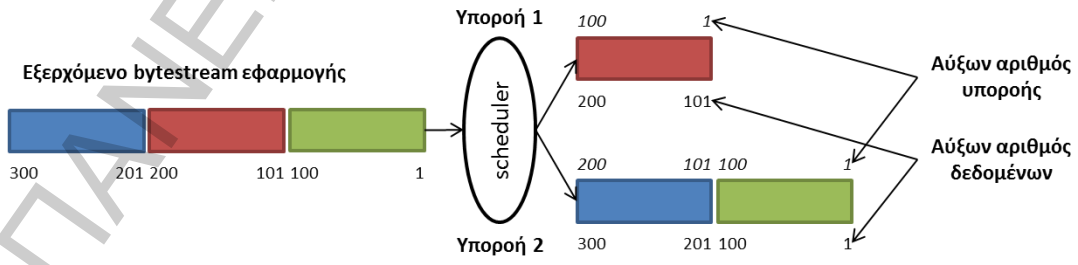
Ανά πάσα στιγμή οποιοδήποτε σύστημα εκ των δύο μπορεί να εγκαταστήσει μια νέα υποροή, επαναλαμβάνοντας εκ νέου την διαδικασία της τριμερής χειραψίας. Εντούτοις, το TCP Option που χρησιμοποιείται για την προσθήκη υποροών διαφέρει. Το σύστημα πρέπει να είναι σε θέση να κατανοήσει ότι η νέα υποροή πρέπει να συμπεριληφθεί στην υφιστάμενη σύνδεση Multipath TCP, και αυτό επιτυγχάνεται με κάποιο διακριτικό (token). Κατά τη διάρκεια της τριμερής χειραψίας, τόσο το

Σύστημα Α όσο και το Σύστημα Β επιλέγουν ένα διακριτικό για την αναγνώριση των νέων συνδέσεων τοπικά. Δεδομένου ότι το Σύστημα Α έχει δύο διευθύνσεις, μπορεί να εγκαταστήσει μια νέα υποροή μεταξύ της διεύθυνσης Α.2 και Β.1, προσκομίζοντας στο Σύστημα Β το διακριτικό Β (token_B) εντός του TCP Option MP_JOIN. Σημειώνεται δε ότι το διακριτικό Α (token_A) δεν απαιτείται να συμπεριληφθεί στη μήνυμα SYN+ACK λόγω ότι το Σύστημα Α γνωρίζει ήδη την κατάσταση της συγκεκριμένης υποροής. Ομοίως, το Σύστημα Β μπορεί να εγκαταστήσει μια νέα υποροή μεταξύ της διεύθυνσης Α.1 και Β.2. Παρόλα αυτά, κανένα από τα δύο συστήματα μπορεί να εγκαταστήσει μια νέα υποροή μεταξύ των διευθύνσεων Α.2 και Β.2, επειδή κανένα από τα δύο συστήματα δεν γνωρίζει για την ύπαρξη της δεύτερης διεπαφής. Συνεπώς, απαιτείται ένα νέο TCP Option, το οποίο ονομάζεται ADD_ADDRESS. Η χρήση της απεικονίζεται στο δεξί μέρος του Σχήμα 7. Μετά την γνωστοποίηση εκ μέρους του Συστήματος Β ότι μπορεί να προσπελαστεί μέσω της διεύθυνσης Β.2, το σύστημα Α έχει την δυνατότητα να εγκαταστήσει ακόμα μια υποροή μεταξύ Α.2 και Β.2.

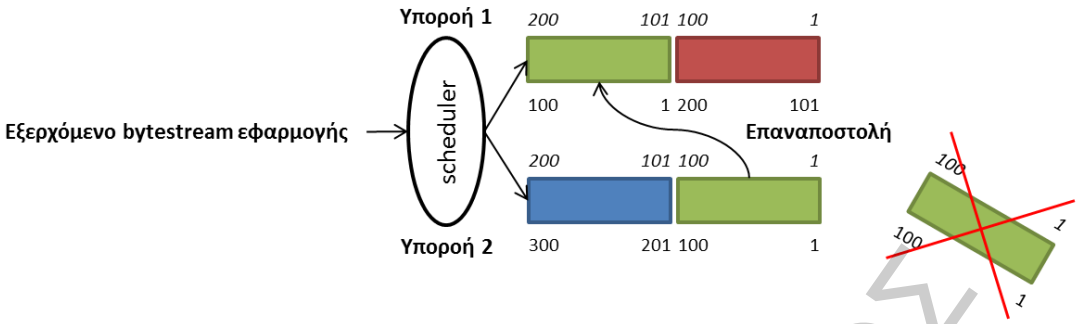
Τέλος, σημειώνεται ότι οι υποροές MPTCP μπορούν να εγκατασταθούν χρησιμοποιώντας οποιαδήποτε έκδοση του Πρωτοκόλλου Διαδικτύου (IPv4 ή IPv6). Αυτή είναι μια ιδιαίτερα σημαντική ιδιότητα του Multipath TCP, καθώς διευκολύνει την μετάβαση από το IPv4 στο IPv6 και επιτρέπει σε συστήματα που διαθέτουν και τις δύο εκδόσεις να τις χρησιμοποιούν ταυτόχρονα.

2.4 Ανταλλαγή δεδομένων σε πολλαπλές ροές

Από την πλευρά του αποστολέα, ένας προγραμματιστής (scheduler), του οποίου ο αλγόριθμος διανομής διαφέρει από την εκάστοτε υλοποίηση, αποφασίζει μέσω ποιας από τις διαθέσιμες υποροές θα αποσταλεί το επόμενο πακέτο δεδομένων (Σχήμα 8 και Σχήμα 9). Ως εκ τούτου, η αρχική σειρά της ροής των bytes έχει απολεσθεί, με αποτέλεσμα να εμφανίζονται κενά στους αριθμούς ακολουθίας της ροής, όπως απεικονίζεται στην υποροή 2 του Σχήμα 8. Αυτό όμως δεν είναι επιθυμητό, καθώς μερικές συσκευές δικτύου, όπως είναι για παράδειγμα οι κανονικοποιητές TCP [HPK01], αναλύουν τους αριθμούς ακολουθίας του TCP και μπλοκάρουν μια ροή TCP όταν εμφανίζονται κενά σε αυτές. Γι' αυτό το λόγο προέκυψε ένας δεύτερος χώρος αριθμών ακολουθίας. Ένας χώρος αριθμών ακολουθίας σχετίζεται με τους αριθμούς ακολουθίας της υποροής. Κάθε υποροή διατηρεί το δικό της χώρο αριθμών ακολουθίας, και τα σημειώνει στο ίδιο πεδίο με εκείνο που χρησιμοποιείται στο TCP. Ο δεύτερος χώρος αριθμών ακολουθίας αποθηκεύεται σε ένα TCP Option, ονόματι DSS (Data Sequence Signal), που έχει οριστεί στο [FRHB12] για την μεταφορά των αριθμών ακολουθίας δεδομένων.



Σχήμα 8: Αριθμοί Ακολουθίας Δεδομένων (Data Sequence Numbers - DNS) του πρωτοκόλλου MPTCP



Σχήμα 9: Επανεκπομπές στο πρωτόκολλο MPTCP

Στα Σχήματα Σχήμα 8 και Σχήμα 9, οι αριθμοί ακολουθίας της υπορροής αναγράφονται στο πάνω μέρος της ροής, ενώ οι αριθμοί ακολουθίας δεδομένων αναγράφονται στο κάτω μέρος της ροής.

Επανεκπομπές: Όταν εντοπίζεται μια απώλεια (ο εντοπισμός απωλειών πραγματοποιείται σε κάθε υπορροή), δεν είναι απαραίτητα η καλύτερη επιλογή η επανεκπομπή του πακέτου στην ίδια υπορροή. Για παράδειγμα, αν η απώλεια συμβεί σε ένα μονοπάτι με υψηλή καθυστέρηση, θα ήταν ίσως συνετό να επανεκπεμφθεί σε άλλη υπορροή. Αυτό μπορεί να επιτευχθεί με το Multipath TCP, επανασυσχετίζοντας τους αριθμούς ακολουθίας δεδομένων με τους αριθμούς ακολουθίας της νέας υπορροής, όπως απεικονίζεται στο Σχήμα 9. Ωστόσο, το αρχικό τμήμα θα πρέπει να επανεκπέμπεται από την αρχική υπορροή, ώστε να αποφεύγεται η σύγχυση που προκαλείται στους ενδιάμεσους κόμβους (middleboxes), όταν εκείνα ελέγχουν για την συνέπεια του ωφέλιμου φορτίου κατά την επανεκπομπή.

Επιβεβαιώσεις: Λόγω του γεγονότος ότι οι υπορροές συμπεριφέρονται κατά τον ίδιο τρόπο με τις κανονικές ροές TCP, οι αριθμοί ακολουθίας υπορροής επιβεβαιώνονται κανονικά σε κάθε υπορροή. Θεωρητικά αυτό θα πρέπει να επαρκή, επειδή ο αποστολέας μπορεί να συμπεράνει τους επιβεβαιωμένους αριθμούς ακολουθίας

δεδομένων από την ληφθείσα επιβεβαίωση της υποροής. Εντούτοις, υπάρχουν δύο πρακτικά προβλήματα με αυτή τη λύση:

- Έχουν αναπτυχθεί διαφόρων ειδών ενδιάμεσοι κόμβοι (middleboxes), για την βελτίωση της συμπεριφοράς του TCP, που καλούνται Διαμεσολαβητές Ενίσχυσης Απόδοσης (Performance Enhancing Proxies – PEP) [BKG+01]. Μερικοί διαμεσολαβητές επιβεβαιώνουν την λήψη των δεδομένων προτού επιβεβαιωθούν από το σύστημα που προορίζονται. Στην περίπτωση όπου απολεσθούν τα δεδομένα μετά την επιβεβαίωση τους από τους ΔΕΑ, τότε ο διαμεσολαβητής είναι υπεύθυνος για την επανεκπομπή τους. Ωστόσο, αν το μονοπάτι έχει αποτύχει, το οποίο σημαίνει ότι αποτυγχάνει κάθε προσπάθεια επανεκπομπής του ΔΕΑ, οι άλλες υποροές δεν μπορούν να χρησιμοποιηθούν για την επανεκπομπή του απολεσθέντος τμήματος, επειδή ο αποστολέας δεν διαθέτει πλέον το τμήμα δεδομένων στην μνήμη.

- Στο κανονικό TCP, το παράθυρο λήψης καθορίζεται ως ακολούθως [Pos81b]:

  [RCV.NXT, RCV.NXT + RCV.WND]

  Η μεταβλητή *RCV.NXT* καθορίζεται από το πεδίο *ACK* στο τμήμα του TCP. Εντούτοις, στο Multipath TCP δεν καθορίζεται κάποιο συγκεκριμένο παράθυρο λήψης ανά υποροή. Η κοινοποίηση του παραθύρου λήψης σχετίζεται με το χώρο αριθμού ακολουθίας δεδομένων, και καθορίζεται εκ νέου ως ακολούθως:

  [DATA.RCV.NXT, DATA.RCV.NXT + DATA.RCV.WND]

Το παράθυρο λήψης παύει πλέον να είναι μέρος του χώρου αριθμών ακολουθίας υποροής και γίνεται μέρος του χώρου αριθμών ακολουθίας δεδομένων. Αν η επιβεβαίωση των δεδομένων προκύπτει από την επιβεβαίωση της υποροής, τότε αυτή δεν είναι συσσωρευτική και συνεπώς δεν αντικατοπτρίζεται στο *DATA.RCV.NXT*, που είναι ο επόμενος αριθμός ακολουθίας δεδομένων που αναμένεται από τον παραλήπτη.

Για την επίλυση των προβλημάτων αυτών, οι προδιαγραφές του πρωτοκόλλου καθορίζουν ένα TCP Option επιβεβαίωσης δεδομένων (data acknowledgement option), το οποίο περιλαμβάνεται στο ίδιο TCP Option με τον αριθμό ακολουθίας δεδομένων. Αυτό το TCP Option επιλύει το πρώτο πρόβλημα, καθώς η συσσωρευτική επιβεβαίωση δεδομένων μπορεί να αποσταλεί από οποιαδήποτε υποροή, και αντικατοπτρίζει την πραγματική κατάσταση του παραλήπτη, ακόμα και στην παρουσία ΔΕΑ. Επίσης επιλύει το δεύτερο πρόβλημα, επειδή καθορίζει ρητά την αρχή (αριστερή άκρη) του παραθύρου λήψης δεδομένων. Τέλος, απλοποιεί ακόμα και τις υλοποιήσεις, αφαιρώντας την ανάγκη αναγνώρισης των επιβεβαιώσεων δεδομένων από τις επιβεβαιώσεις σε επίπεδο υποροής.

Έλεγχος Συμφόρησης: Οι αλγόριθμοι ελέγχου συμφόρησης για το κανονικό TCP προσπαθούν να διανέμουν με δίκαιο τρόπο την διαθέσιμη χωρητικότητα. Δύο υποροές TCP, που ανήκουν στην ίδια λογική σύνδεση, θα χρησιμοποιούσαν την διπλάσια χωρητικότητα από εκείνη που δικαιούνται. Επιπλέον, σε έρευνα των [WRGH11] περιγράφεται ένα σενάριο με πολλαπλά σημεία συμφόρησης, όπου ακόμα και ένα «επιθετικό» Multipath TCP θα αποτύχανε να λάβει το καλύτερο δυνατό εύρος ζώνης. Αυτό είναι επιθυμητό, σε σενάριο πολυδιαδρομικότητας , ώστε να χρησιμοποιούν μόνο τα μονοπάτια όπου παρατηρείται λιγότερη συμφόρηση, αντί της εξάπλωσης της κίνησης εξίσου μεταξύ των διαθέσιμων μονοπατιών. Ξεκινώντας

από μια υπάρχουσα θεωρητική λύση [KV05, HSH+06], αναπτύχθηκε ένας αλγόριθμός που εκπληρώνει τους δύο παρακάτω στόχους:

- Μια πολυδιαδρομική ροή θα πρέπει να δίδει μια σύνδεση με απόδοση (throughput) τουλάχιστον ίση με εκείνη μιας μονοδρομικής ροής TCP, χρησιμοποιώντας το καλύτερο από τα προσφερόμενα μονοπάτι. Αυτό εξασφαλίζει ότι υπάρχει ένα κίνητρο για την ανάπτυξη πολυδιαδρομικών ροών.
- Μια πολυδιαδρομική ροή δεν πρέπει να λαμβάνει μεγαλύτερη χωρητικότητα σε κανένα μονοπάτι ή ομάδα μονοπατιών σε σχέση με μια μονοδρομική ροή TCP, χρησιμοποιώντας το καλύτερο από τα προσφερόμενα μονοπάτι. Αυτό εγγυάται ότι δεν θα βλάψει αδικαιολόγητα άλλες ροές σε μια συμφορημένη ζεύξη, δίχως να έχει σημασία ποιος συνδυασμό μονοπατιών περνά μέσω αυτής της ζεύξης.

Ο αλγόριθμός ελέγχου συμφόρησης που προκύπτει, έχει ως ακολούθως:

- Για κάθε μη-διπλή επιβεβαίωση στην υποροή i, το παράθυρο συμφόρησης της υποροής i αυξάνεται κατά:

$$\min\left(\frac{\alpha * bytes\_acked * mss_i}{cwnd_{tot}}, \frac{bytes\_acked * mss_i}{cwnd_i}\right),$$

όπου $cwnd_{tot}$ είναι το συνολικό παράθυρο συμφόρησης όλων των υποροών και

$$\alpha = cwnd_{tot} \frac{max_i\left(\frac{cwnd_i}{RTT_i^2}\right)}{\left(\sum_i \frac{cwnd_i}{RTT_i}\right)^2}$$

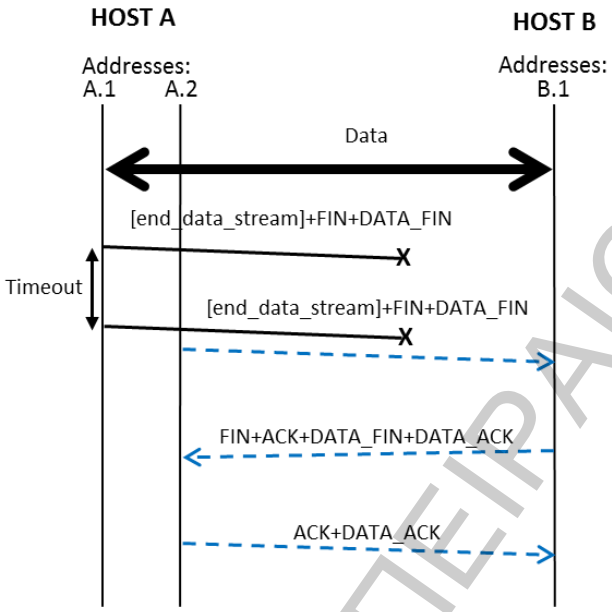
Ο συγκεκριμένος τύπος προϋποθέτει ότι το παράθυρο συμφόρησης μετριέται σε bytes και η μεταβλητή *bytes_acked* είναι ο αριθμός των bytes που έχουν επιβεβαιωθεί από το ληφθέν τμήμα επιβεβαίωσης.

- Κατά τον εντοπισμό της απώλειας στην υποροή i, το παράθυρο συμφόρησης της υποροής μειώνεται κατά $cwnd_i/2$

2.5 Τερματισμός σύνδεσης στο MultiPath TCP

Θα μπορούσε κανείς να υποθέσει, ότι για τον τερματισμό μιας σύνδεσης Multipath TCP, θα αρκούσε η αποστολή ενός κανονικού μηνύματος `FIN` σε όλες τις υποροές. Αυτό θα επαρκούσε μόνο στην περίπτωση όπου όλες οι υποροές λειτουργούν κανονικά, αλλά θα αποτύγχανε στην περίπτωση όπου μια τουλάχιστον υποροή δεν είναι λειτουργική. Στο TCP, αν μια ροή δεν είναι λειτουργική, τότε η σύνδεση δεν θα μπορούσε να τερματιστεί ομαλώς. Ωστόσο, το Multipath TCP δεν θα επηρεαζόταν από μια τέτοια αποτυχία, επειδή κάθε χρήσιμη πληροφορία για την σύνδεση μπορεί να μετακινηθεί σε άλλη λειτουργική υποροή.

Το μήνυμα `Data FIN` έχει την ίδια σημασία με το μήνυμα `FIN` στο επίπεδο δεδομένων. Ένα κανονικό μήνυμα `FIN` δεν τερματίζει μια σύνδεση, αλλά μια υποροή. Ένα μήνυμα `Data FIN` μπορεί να αποσταλεί σε οποιαδήποτε υποροή. Επιβεβαιώνεται με ένα μήνυμα `Data ACK`, και καταλαμβάνει 1 byte στο χώρο αριθμού ακολουθίας δεδομένων.



Σχήμα 10: Παράδειγμα τερματισμού σύνδεσης στο MPTCP

Ένα παράδειγμα τερματισμού σύνδεσης απεικονίζεται στο Σχήμα 10. Στην συγκεκριμένη περίπτωση, η υποροή μεταξύ των A.1 και B.1 έχει αποτύχει. Ο όρος `end_data_stream` αναπαριστά το τελευταίο μπλοκ δεδομένων της εφαρμογής. Δεδομένου ότι η εφαρμογή έχει καλέσει την κλήση συστήματος `close()` μετά το τελευταίο μπλοκ δεδομένων προς αποστολή, το Multipath TCP επισυνάπτει ένα μήνυμα `Data FIN` στην ουρά αποστολής. Έστω ότι ο scheduler επιλέγει την υποροή A.1 – B.1. Δεδομένης της παρουσίας του μηνύματος `Data FIN` στην ουρά αποστολής, το Multipath TCP γνωρίζει ότι μπορεί να τερματίσει την υποροή, και θέτει επίσης το μήνυμα `FIN` . Το τμήμα τερματισμού χάνεται, και μετά από κάποιο timeout επανεκπέμπεται. Το χαμένο τμήμα επανεκπέμπεται επίσης στην υποροή A.2 – B.1, χάρη στην ικανότητα του Multipath TCP να επανεκπέμπει τμήματα σε άλλες υποροές. Τέλος, το τμήμα τερματισμού φθάνει στον προορισμό και ο τερματισμός

μπορεί να ολοκληρωθεί ομαλά στην δεύτερη υποροή. Από την πλευρά της εφαρμογής, η στοίβα μπορεί να επιστρέψει ένα μήνυμα, το οποίο να πληροφορεί για το επιτυχή τερματισμό. Από την πλευρά του Multipath TCP, η αποτυχημένη υποροή θα συνεχίσει την επαναποστολή των τμημάτων `FIN` και τελικά να τερματιστεί μετά από κάποιο timeout. Αυτή η διαδικασία γίνεται χωρίς να την γνωρίζει το επίπεδο εφαρμογής. Σημειώνεται ότι στην έρευνα [FRHB12] προτείνεται η μείωση της τιμής του timeout όταν η υποροή δεν χρειάζεται πλέον για την σύνδεση.

2.6 Μηχανισμοί ασφάλειας για το MultiPath TCP

Στο [Bag11] πραγματοποιείται μια ανάλυση απειλών για το Multipath TCP. Αναφέρονται δύο σημαντικά θέματα σχετικά με το Multipath TCP, τα οποία είναι:

- Ένας σημαντικός στόχος σχεδίασης του Multipath TCP είναι να μην είναι λιγότερο ασφαλές από το TCP. Για παράδειγμα, το TCP είναι ευάλωτο στις επιθέσεις τύπου «Man-in-the-middle», οπότε δεν αποτελεί στόχος του Multipath TCP να προστατεύεται ενάντια σε αυτές τις επιθέσεις.

- Η κύρια νέα απειλή που εισήγαγε το Multipath TCP σχετίζεται με την ευελιξία εντοπισμού. Ένας εισβολέας θα μπορούσε να στείλει ένα μήνυμα `ADD_ADDRESS` με την δική του διεύθυνση και να ανακατευθύνει μέρος της ροής δεδομένων, αν όχι ολόκληρης, σε εκείνον. Διαφορετικά, θα μπορούσε να

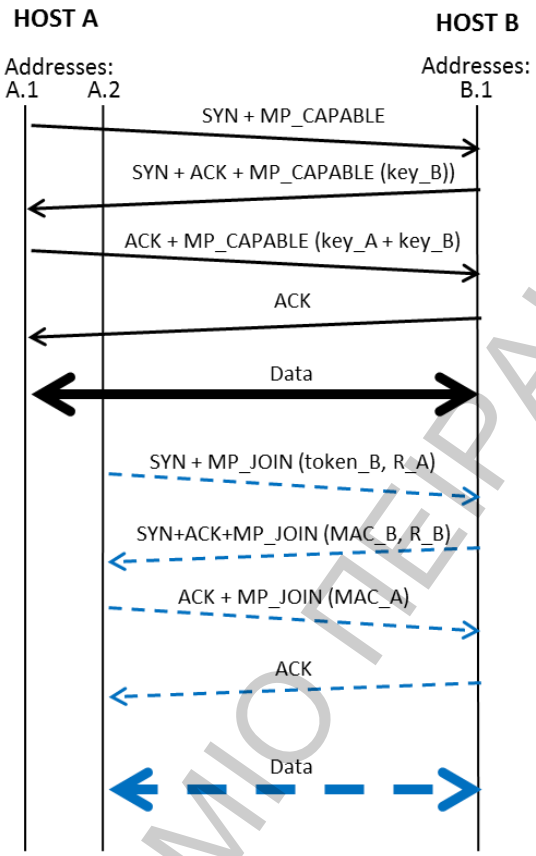
στείλει ένα μήνυμα `ADD_ADDRESS` με την διεύθυνση ενός στόχου, ούτως ώστε να αναγκάσει τον εξυπηρετητή να επιτεθεί σε ένα επιλεγμένο θύμα.

Από τα παραπάνω, προκύπτει ότι το πιο σημαντικό μέρος του Multipath TCP, από άποψη ασφάλειας, είναι η εγκατάσταση υποροών. Η έρευνα [Bag11] τονίζει ότι θα πρέπει να υπάρχει μια ανταλλαγή (trade-off) μεταξύ του επιπέδου προστασίας που επιθυμείται και την πολυπλοκότητα των μηχανισμών ασφαλείας που προκύπτουν. Η ανταλλαγή (trade-off) που υιοθετήθηκε, όπως ορίζεται στην έρευνα των [FRHB12], βασίστηκε στην παραδοχή ότι για την προστασία των νέων υποροών, η αρχική τριμερής χειραψία δεν μπορεί να υποκλαπεί από έναν εισβολέα. Άλλοι μηχανισμοί ασφαλείας θεωρήθηκαν ιδιαίτερα δαπανηροί σε υπολογιστικούς πόρους, και ως εκ τούτου δεν δίνεται καμία προστασία ενάντια σε ετεροχρονισμένες επιθέσεις που περιγράφονται στην έρευνα [Bag11], όπου ένας εισβολέας είναι στο μονοπάτι από την αρχή της σύνδεσης και στη συνέχεια απομακρύνεται, αλλά έχει ακόμα την δυνατότητα να καταλάβει ακόμα την σύνδεση. Το πρωτόκολλο TCP θα πρέπει να επεκταθεί ώστε να υποστηριχθούν άλλοι μηχανισμοί ασφαλείας που θα αναπτυχθούν στο μέλλον.

Με βάση την παραδοχή ότι ένας εισβολέας δεν μπορεί να δει την αρχική τριμερή χειραψία, ανταλλάσσεται ένα μη-κρυπτογραφημένο κλειδί μεταξύ των μερών. Το Σύστημα Α παράγει ένα κλειδί Α και αντίστοιχα το Σύστημα Β παράγει ένα κλειδί Β. Το κλειδί Α αποστέλλεται μόνο στο τρίτο μέρος της χειραψίας, για να επιτραπεί η χρήση των SYN cookies στον εξυπηρετητή. Σημειώνεται δε ότι το διακριτικό (token), δεν εμφανίζεται πλέον στην αρχική ανταλλαγή, όπως παρουσιάστηκε στο Σχήμα 7. Αυτό οφείλεται στο γεγονός ότι το διακριτικό παράγεται από το κλειδί:

$$token_A = hash(key_A)$$

$$token_B = hash(key_B)$$



Σχήμα 11: Πιστοποίηση στο MPTCP

Το κλειδί στη συνέχεια χρησιμοποιείται για την αυθεντικοποίηση των εγκαταστάσεων νέων υποροών που θα προκύψουν στη συνέχεια της σύνδεσης, χάρη σε έναν Κωδικό Αυθεντικοποιήσης Μηνύματος βασισμένο σε Αλγόριθμο Κατακερματισμού (Hash-based Message Authentication Code – HMAC), με την συνάρτηση κατακερματισμού να είναι η SHA1 [EH06]. Λόγω του γεγονότος ότι οι διευθύνσεις και οι πόρτες μπορούν να αλλάξουν κατά τη πορεία προς τον προορισμό, για παράδειγμα από μια συσκευή NAPT [SE01], αυτά δεν μπορούν να συμπεριληφθούν στην αυθεντικοποίηση HMAC. Αντί αυτού, ορίζεται ένα ζευγάρι τυχαίων αριθμών, R_A και R_B, τα οποία παράγονται τοπικά και από το ομότιμο

σύστημα αντίστοιχα, που πρέπει να χρησιμοποιηθούν ως το μήνυμα που πρέπει να κατακερματιστεί. Το Σύστημα Α αυθεντικοποιεί τον εαυτό του, δείχνοντας ότι είναι σε θέση να παρέχει το σωστό κατακερματισμό των κλειδιών R_A και R_B, χρησιμοποιώντας τα κλειδιά Α και Β. Ομοίως, το Σύστημα Β δείχνει ότι είναι σε θέση να παράξει το σωστό HMAC βασισμένο στα R_A και R_B χρησιμοποιώντας τα κλειδιά Α και Β. Δεδομένου ότι μόνο τα Συστήματα Α και Β γνωρίζουν τα κλειδιά, τότε μόνο εκείνα μπορούν να δημιουργήσουν νέες υποροές μεταξύ τους, δεδομένου ότι και τα τυχαία κλειδιά είναι αρκετά μεγάλα σε μέγεθος για να αποτρέψει μια επίθεση «ωμής βίας» (brute force attack). Επιπλέον, για την προστασία από επιθέσεις αναπαραγωγής, οι αριθμοί R_A και R_B χρησιμοποιούνται μόνο μια φορά.

Στο Σχήμα 11απεικονίζονται οι χειραψίες μεταξύ των μερών, οι οποίες αριθμούν πλέον σε 4, τόσο για την αρχική υποροή όσο και για τις πρόσθετες. Στην πραγματικότητα, η τριμερής χειραψία που πραγματοποιείται στο TCP διατηρείται, αλλά το Multipath TCP αναγκάζει τον εξυπηρετητή να αποστείλει μια επιβεβαίωση μετά την τριμερή χειραψία για να εξασφαλίσει ότι τα τελικά δεδομένα ασφαλείας του Multipath TCP έχουν ληφθεί. Για την αρχική ανταλλαγή, τα δεδομένα είναι τα κλειδιά Α και Β, ενώ για τις πρόσθετες υποροές είναι το MAC_A (Message Authentication Code).

ΚΕΦΑΛΑΙΟ 3

ΑΝΑΛΥΣΗ ΤΗΣ ΥΛΟΠΟΙΗΣΗΣ ΤΟΥ MULTIPATH TCP ΣΤΟ LINUX

Το πρωτόκολλο Multipath TCP [FRHB12] είναι μια από τις μεγαλύτερες επεκτάσεις του πρωτοκόλλου TCP που επιτρέπει την ταυτόχρονη χρήση πολλαπλών μονοπατιών σε δύο ή και περισσότερες διεπαφές. Επίσης είναι μη αντιληπτό ως προς το επίπεδο εφαρμογής, πράγμα που σημαίνει ότι δεν απαιτείται καμία αλλαγή στο επίπεδο εφαρμογής που χρησιμοποιεί το πρωτόκολλο TCP, και είναι δίκαιο ως προς τις υπόλοιπες ροές TCP [RHW11]. Οι στόχοι του σχεδιασμού και η αρχιτεκτονική του πρωτοκόλλου Multipath TCP αναλύονται στο [FRH+11]. Εκτός από την αρχιτεκτονική του πρωτοκόλλου, θα πρέπει να γίνουν κάποιες επιλογές σχεδιασμού προκειμένου να επεκταθεί μια υφιστάμενη υλοποίηση TCP για την υποστήριξη του Multipath TCP.

Η προτεινόμενη αρχιτεκτονική αναμένεται να ισχύσει ανεξάρτητα από το λειτουργικό σύστημα, παρόλο που η υλοποίηση του πρωτοκόλλου έχει πραγματοποιηθεί για το λειτουργικό σύστημα Linux. Ακόμα ένας στόχος είναι η επίτευξη όσο το δυνατόν μεγαλύτερης επεκτασιμότητας χωρίς να έχει αρνητικό αντίκτυπο στην αποδοτικότητα και ως εκ τούτου να επιτρέπει την συνύπαρξη άλλων πολυδιαδρομικών πρωτοκόλλων.

Η ανάλυση της υλοποίησης του πρωτοκόλλου Multipath TCP βασίστηκε στην έκδοση 0.6 της υλοποίησης. Η έκδοση αυτή περιλαμβάνει γύρω στις 10.000 γραμμές κώδικα και η εκτέλεση γίνεται στο χώρο του πυρήνα (kernel-space αντί για user-space).

3.1 Ορολογία

Για την κατανόηση της υλοποίησης του Multipath TCP στο Linux, παρατίθενται οι ακόλουθοι όροι:

- **Meta-socket:** Μια δομή socket που χρησιμοποιείται για την ανακατάταξη των εισερχομένων δεδομένων στο επίπεδο σύνδεσης και για τον προγραμματισμό των εξερχομένων δεδομένων προς τις υποροές.

- **Master subsocket:** Η δομή socket που επικοινωνεί άμεσα με το επίπεδο εφαρμογής. Στην περίπτωση που χρησιμοποιείται το κανονικό πρωτόκολλο TCP, το master subsocket είναι το μόνο ενεργό socket. Όταν χρησιμοποιείται

το πρωτόκολλο Multipath TCP, αυτό το socket αναπαριστά την πρώτη υποροή.

- **Slave subsocket:** Κάθε socket που δημιουργείται από τον πυρήνα για την παροχή πρόσθετων υποροών. Αυτά τα sockets αποκρύπτονται από το επίπεδο εφαρμογής.

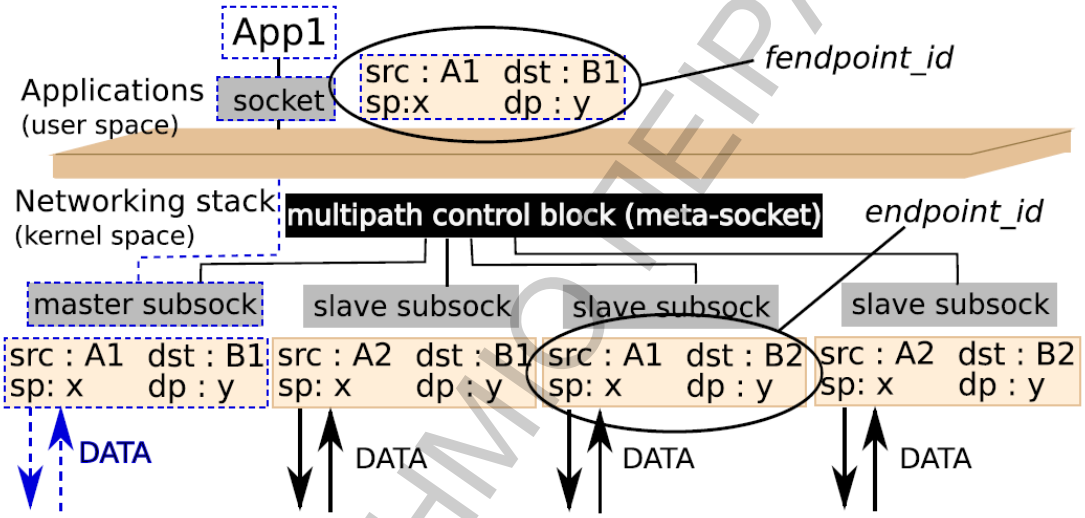
- **Endpoint ID:** Αναγνωριστικό ζεύξης. Είναι ένα σύνολο δεδομένων που προσδιορίζει μια συγκεκριμένη υποροή, και ως εκ τούτου ένα συγκεκριμένο subsocket. Το σύνολο δεδομένων αποτελείται από τις διευθύνσεις IP (IP address) και τις πόρτες (ports) του τοπικού και του απομακρυσμένου συστήματος (source, destination) (`saddr, sport, daddr, dport`).

- **Fendpoint ID:** Αναγνωριστικό πρώτης ζεύξης. Αποτελεί το αναγνωριστικό ζεύξης του Master subsocket.

- **Connection ID ή token:** Ένας τοπικός μοναδικός αριθμός, που καθορίζεται στο [FRHB12], που επιτρέπει τον εντοπισμό μιας σύνδεσης κατά τη διάρκεια της εγκατάστασης νέων υποροών.

- **local_addr_table:** Ένας πίνακας με τις τοπικές διευθύνσεις. Αποθηκεύει, ανά σύνδεση, ένα σύνολο τοπικών διευθύνσεων που μπορεί να χρησιμοποιείται το πρωτόκολλο Multipath TCP.

- **remote_addr_table:** Ένας πίνακας με τις απομακρυσμένες διευθύνσεις. Αποθηκεύει, ανά σύνδεση, ένα σύνολο από απομακρυσμένες διευθύνσεις που έχει μάθει το Multipath TCP από τον εταίρο του, είτε από τα μηνύματα ADD_ADDRESS, είτε από μηνύματα SYN από την πλευρά του εταίρου χρησιμοποιώντας νέες διευθύνσεις.



Σχήμα 12: Επισκόπηση της αρχιτεκτονικής του Multipath TCP

3.2 Αρχιτεκτονική για πολυδιαδρομική μετάδοση

Η αρχιτεκτονική του πρωτοκόλλου Multipath TCP αποτελείται από τέσσερις οντότητες:

- την Διαχείριση Μονοπατιών (Path Management)
- τον Προγραμματισμό Πακέτων (Packet Scheduling)

- την Διεπαφή Υποροής (Subflow Interface)
- και τον Έλεγχο Συμφόρησης (Congestion Control)

Αυτές οι οντότητες μπορούν να κατηγοριοποιηθούν περαιτέρω με βάση το επίπεδο στο οποίο λειτουργούν:

- Επίπεδο μεταφοράς: Στην κατηγορία αυτή ανήκει ο Προγραμματισμός Πακέτων, η Διεπαφή Υποροής και ο Έλεγχος Συμφόρησης. Οι τρεις αυτές οντότητες μπορούν να ομαδοποιηθούν υπό τον όρο «Πολυδιαδρομική Μεταφορά» (Multipath Transport – MT). Από άποψη υλοποίησης, αυτό προϋποθέτει τροποποιήσεις στο TCP.

- Κάτω από το επίπεδο μεταφοράς: Στην κατηγορία αυτή ανήκει η Διαχείριση Μονοπατιών. Η Διαχείριση Μονοπατιών μπορεί να πραγματοποιηθεί στο επίπεδο μεταφοράς, όπως είναι η περίπτωση του ενσωματωμένου Διαχειριστή Μονοπατιών που περιγράφεται στο [FRHB12]. Αυτός ο διαχειριστής ανακαλύπτει μονοπάτια μέσα από την ανταλλαγή μηνυμάτων, όπως είναι το TCP Option `ADD_ADDR`, και καθορίζει ένα μονοπάτι ως ένα Endpoint ID.

Λόγω της θεμελιώδους ανεξαρτησίας της διαχείρισης μονοπατιών σε σύγκριση με τις υπόλοιπες τρεις οντότητες, υπάρχει μια σαφής διαχωριστική γραμμή μεταξύ των δύο, και καθορίζει μια απλή διεπαφή που επιτρέπει στο Multipath TCP να επωφεληθεί εύκολα από οποιαδήποτε κατάλληλη διεπαφή πολυδιαδρομικότητας.

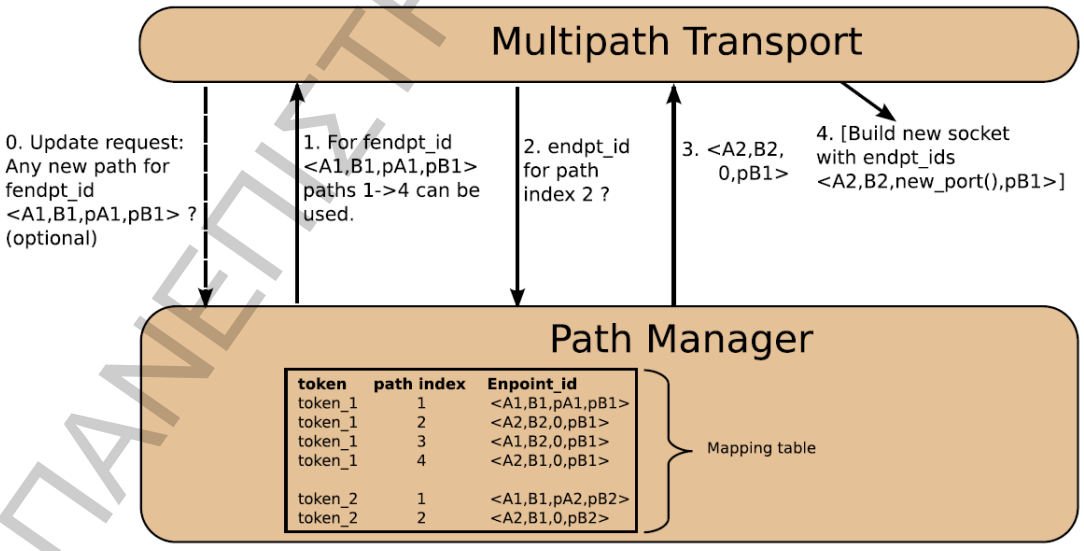
3.2.1 Αρχιτεκτονική του Multipath TCP

Παρόλο που το Multipath TCP εμπεριέχεται πλήρως στο επίπεδο μεταφοράς, εντούτοις μπορεί να οργανωθεί ως ένας Διαχειριστής Μονοπατιών και ένα Επίπεδο Πολυδιαδρομικής Μεταφοράς. Ο Διαχειριστής Μονοπατιών αναφέρει στο επίπεδο Πολυδιαδρομικής Μεταφοράς τα μονοπάτια που μπορούν να χρησιμοποιηθούν για μια σύνδεση Multipath TCP, τα οποία αναγνωρίζονται από ένα fendpoint ID. Το fendpoint ID είναι το σύνολο πληροφοριών, που αποτελείται από τις διευθύνσεις IP (IP address) και τις πόρτες (ports) του τοπικού και του απομακρυσμένου συστήματος (source, destination), όπως εμφανίζεται από το επίπεδο εφαρμογής και προσδιορίζει μοναδικά την Multipath TCP σύνδεση. Ο Διαχειριστής Μονοπατιών διατηρεί την συσχέτιση μεταξύ του αναγνωριστικού μονοπατιού (path index) και του endpoint ID.

Σημειώνεται δε ότι το fendpoint ID αναπαριστά ένα μονοπάτι και ως εκ τούτου ένα συγκεκριμένο endpoint ID. Το fendpoint ID αναπαριστά το πρώτο μονοπάτι (path index 1). Όπως εξηγείται στο [FRG+11], δεν είναι ακόμα ξεκάθαρο πως μια υλοποίηση πρέπει να συμπεριφερθεί στο γεγονός της απώλειας της πρώτης υποροής. Αναμένεται ωστόσο ότι το Master subsocket πρέπει να διατηρείται σε χρήση ως μια διεπαφή προς την εφαρμογή, ακόμα και αν δεν αποστέλλονται πλέον δεδομένα από την συγκεκριμένη υποροή. Επιτρέπει επίσης στο fendpoint ID να παραμείνει σημαντικό καθ' όλη τη διάρκεια της σύνδεσης.

Το Σχήμα 13 απεικονίζει ένα παράδειγμα ακολουθίας αλληλεπιδράσεων μεταξύ του Διαχειριστή Μονοπατιών (Path Manager) και του επιπέδου Πολυδιαδρομικής Μεταφοράς (Multipath Transport). Όταν το επίπεδο ΠΜ εκκινεί μια νέα σύνδεση, καλώντας την κλήση συστήματος `connect()` ή `accept()`, μπορεί να αιτηθεί από

τον ΔΜ να τον ενημερώσει σχετικά με τα πιθανά εναλλακτικά μονοπάτια για την νέα σύνδεση (βήμα 0). Ο ΔΜ μπορεί να ενημερώσει αυθόρμητα το επίπεδο ΠΜ σε οποιαδήποτε στιγμή, συνήθως όμως όταν αλλάξει το σύνολο των μονοπατιών (βήμα 1). Με βάση τις πληροφορίες αυτές, το επίπεδο ΠΜ είναι σε θέση να αποφασίσει αν είναι εφικτή η εγκατάσταση νέων υποροών, και εφόσον είναι εφικτή να αποφασιστεί ο αριθμός των νέων υποροών. Στο Σχήμα 13 αποφασίζει να εγκαταστήσει μόνο μια υποροή και αποστέλλει ένα αίτημα για το endpoint ID στον ΔΜ (βήμα 2). Στο βήμα 3, δίδεται η απάντηση εκ μέρους του ΔΜ, η οποία είναι η <A2, B2, 0, pB1>. Η πόρτα στο τοπικό σύστημα είναι αδιευκρίνιστη (0) για να επιτρέψει στο επίπεδο ΠΜ να εξασφαλίσει την μοναδικότητα του νέου endpoint ID, χάρη στην κλήση συστήματος `new_port()`. Σημειώνεται ότι τα βήματα 1, 2 και 3 δεν απαιτείται να είναι πραγματικά μηνύματα και μπορούν να είναι κλήσεις μεθόδων.



Σχήμα 13: Λειτουργικός διαχωρισμός του Multipath TCP στο επίπεδο μεταφοράς

Τα παρακάτω TCP Options, όπως περιγράφονται στο [FRHB12], διαχειρίζονται από το επίπεδο Πολυδιαδρομικής Μεταφοράς:

- `MULTIPATH CAPABLE (MP_CAPABLE)`: Ενημερώνει την άλλη πλευρά για την υποστήριξη του πρωτοκόλλου Multipath TCP και ανακοινώνει το τοπικό διακριτικό.

- `MP_JOIN/MP_AUTH`: Εκκινεί μια νέα υποροή.

- `DATA SEQUENCE NUMBER (DSN_MAP)`: Αναγνωρίζει την θέση ενός σετ από bytes μέσα σε μια μεταροή.

- `DATA FIN (DFIN)`: Τερματίζει μια σύνδεση.

- `MP_PRIO`: Ζητά από την άλλη πλευρά να αναθεωρήσει την κατάσταση ασφαλείας της υποροής στην οποία εμφανίζεται το συγκεκριμένο TCP Option. Παρόλο που το TCP Option αυτό αποστέλλεται από το επίπεδο ΠΜ, ωστόσο μπορεί να προκληθεί από τον ΔΜ.

- `MP_FAIL`: Υποδεικνύει την αποτυχία του αθροίσματος ελέγχου στο επίπεδο σύνδεσης.

Ο Διαχειριστής Μονοπατιών εφαρμόζει μια συγκεκριμένη τεχνολογία για να δώσει στο επίπεδο ΠΜ την δυνατότητα να χρησιμοποιήσει διάφορα μονοπάτια. Ο ενσωματωμένος Διαχειριστής Μονοπατιών του Multipath TCP χρησιμοποιεί

πολλαπλές διευθύνσεις IPv4 και IPv6 ως μέσο για να επηρεάσει την προώθηση των πακέτων μέσω του Διαδικτύου.

Όταν το επίπεδο ΠΜ εκκινεί μια νέα σύνδεση, επιλέγει ένα διακριτικό (token), το οποίο θα χρησιμοποιηθεί για τον προσδιορισμό της σύνδεσης. Αυτή η διαδικασία είναι απαραίτητη για επιτρέψει σε μελλοντικά μηνύματα `SYN` μελλοντικών υποροών, που εμπεριέχουν το TCP Option MP_JOIN, να συνδεθούν με την σωστή σύνδεση.

Στο Σχήμα 13 απεικονίζεται ένα παράδειγμα ενός πίνακα συσχέτισης. Στο συγκεκριμένο παράδειγμα υπάρχουν δύο ενεργές συνδέσεις Multipath TCP. Η πρώτη σύνδεση αναγνωρίζεται με το διακριτικό `token_1`, και η δεύτερη με το διακριτικό `token_2`. Όπως ορίζεται στο [FRHB12], τα διακριτικά πρέπει να είναι μοναδικό στο τοπικό σύστημα.

Δεδομένου ότι το endpoint ID μπορεί να αλλάξει από την μία υποροή στην άλλη, η σύνδεση της εισερχόμενης νέας υποροής, που αναγνωρίζεται από τα TCP Options `SYN` και `MP_JOIN`, με την σωστή σύνδεση επιτυγχάνεται χάρη στο τοπικά μοναδικά διακριτικά.

Τα παρακάτω TCP Options, όπως ορίζονται στο [FRHB12], περιλαμβάνονται στον Διαχειριστή Μονοπατιών:

- Add Address (`ADD_ADDR`): Κοινοποιεί μια νέα διεύθυνση του συστήματος
- Remove Address (`REMOVE_ADDR`): Αποσύρει μια κοινοποιημένη διεύθυνση.

3.2.2 Δομή του Multipath Transport

Το επίπεδο Πολυδιαδρομικής Μεταφοράς χειρίζεται τριών ειδών sockets, που είναι τα εξής:

- **Master subsocket:** Αυτό είναι το πρώτο socket που χρησιμοποιείται στην εκκίνηση μιας σύνδεσης, είτε είναι TCP είτε Multipath TCP. Είναι επίσης το μοναδικό είδος socket που χρησιμοποιείται στην περίπτωση όπου η άλλη πλευρά δεν υποστηρίζει το πρωτόκολλο Multipath TCP. Το socket αυτό αρχικοποιείται από την εφαρμογή μέσω της κλήσης συστήματος `socket()`. Αμέσως μετά την δημιουργία του νέου master subsocket, ενεργοποιείται το Multipath TCP με την δημιουργία του meta-socket.

- **Meta-socket:** Κατέχει το μπλοκ ελέγχου των υποροών και λειτουργεί ως το socket της σύνδεσης προς το επίπεδο εφαρμογής. Η οντότητα αυτή, ως η πηγή των δεδομένων, κατέχει την κύρια μνήμη (buffer) αποστολής. Επίσης διατηρεί την ουρά λήψης σε επίπεδο σύνδεσης και την ουρά λήψης εκτός σειράς, που χρησιμοποιείται για την αναδιάταξη των τμημάτων των δεδομένων. Το meta-socket αναπαριστάται ως μια δομή ενός κανονικού socket (με κάποιες επεκτάσεις) στην υλοποίηση του πρωτοκόλλου Multipath TCP, επειδή επιτρέπει την επαναχρησιμοποίηση ενός μεγάλου μέρους του υφιστάμενου κώδικα του TCP με μερικές τροποποιήσεις. Συγκεκριμένα, η δομή ενός κανονικού socket ήδη περιλαμβάνει τις μεταβλητές κατάστασης `SND.UNA`, `SND.NXT`, `SND.WND`, `RCV.NXT`, `RCV.WND`. Τέλος, κατέχει όλες τις απαραίτητες ουρές για την αποστολή και την λήψη των δεδομένων.

- **Slave subsocket:** Οποιαδήποτε υποροή δημιουργείται από το Multipath TCP, με εξαίρεση την πρώτη υποροή που είναι το master subsocket. Το slave subsocket δημιουργείται από τον πυρήνα και δεν είναι εμφανές από την εφαρμογή. Το master subsocket και τα slave subsockets σχηματίζουν το σύνολο των διαθέσιμων υποροών που ο Προγραμματιστής Πακέτων (Packet Scheduler) του Multipath TCP χρησιμοποιεί για την αποστολή των πακέτων.

3.2.3 Δομή για τον Path Manager

Σε αντίθεση με το επίπεδο Πολυδιαδρομικής Μεταφοράς, το οποίο είναι πιο περίπλοκο και διαιρείται σε επιμέρους οντότητες (Προγραμματιστής Πακέτων, Διεπαφή Υποροών και Έλεγχος Συμφόρησης), ο Διαχειριστής Μονοπατιών διατηρεί μόνο έναν πίνακα συσχέτισης (διακριτικό/token, αναγνωριστικού μονοπατιού, endpoint ID) και ενημερώνει το επίπεδο Πολυδιαδρομικής Μεταφοράς όταν τροποποιηθούν τα περιεχόμενα του πίνακα. Ο Πίνακας 2 περιέχει την συσχέτιση των συμβάντων και των ενεργειών, που πραγματοποιούνται στο σύστημα, που εμπεριέχονται στον Διαχειριστή Μονοπατιών που έχει υλοποιηθεί. Στο παράδειγμα του Σχήμα 13, απεικονίζεται η εκκίνηση της διαδικασίας της εγκατάστασης μιας νέας σύνδεσης Multipath TCP, όπου ο Διαχειριστή Μονοπατιών ενημερώνει το επίπεδο Πολυδιαδρομικής Μεταφοράς σχετικά με τον αριθμό των αξιοποιήσιμων μονοπατιών. Όταν το επίπεδο Πολυδιαδρομικής Μεταφοράς ζητά πληροφορίες για το endpoint ID του μονοπατιού με ID 2, ο Διαχειριστής Μονοπατιών δίδει την απάντηση < A2, B2, 0, pB1 >. Η μηδενική τιμή που αναφέρεται στην πόρτα πηγής (source port) σημαίνει ότι ο Διαχειριστής Μονοπατιών δεν υποδεικνύει κάποια συγκεκριμένη

πόρτα, με αποτέλεσμα η πόρτα να επιλέγεται από το επίπεδο Πολυδιαδρομικής Μεταφοράς, χρησιμοποιώντας τον υφιστάμενο αλγόριθμο επιλογής πόρτας του πρωτοκόλλου TCP.

ΣΥΜΒΑΝ	ΕΝΕΡΓΕΙΑ
Δέσμευση master_sk : Ενεργοποιείται από την κλήση συστήματος `bind()` ή `connect()`, ή όταν εγκαθίσταται ένα νέο socket από την πλευρά του εξυπηρετητή	Ανακαλύπτει το σύνολο των τοπικών διευθύνσεων. Αυτό το σύνολο στη συνέχεια διατηρείται στην δομή local_addr_table.
Λήψη των `ADD_ADDR` ή `SYN + MP_JOIN` σε νέα διεύθυνση	Ενημερώνεται ο πίνακας remote_addr_table αντίστοιχα
Ενημέρωση των πινάκων **local/remote_addr_table**	Ενημερώνεται η δομή του πίνακα mapping_table με τις προσθήκη οποιασδήποτε νέας διεύθυνσης ή αφαιρούνται εκείνες που έχουν τερματιστεί. Σε κάθε ζεύγος διευθύνσεων αποδίδεται ένα αναγνωριστικό (ID) μονοπατιού
Ενημέρωση του **mapping_table**	Αποστολή ειδοποίησης στο επίπεδο Πολυδιαδρομικής Μεταφοράς.
Λήψη από το επίπεδο Πολυδιαδρομικής Μεταφοράς αιτήματος Endpoint ID	Λήψη του endpoint ID για το αντίστοιχο ID μονοπατιού από τον πίνακα συσχέτισης και επιστροφή του αποτελέσματος στο επίπεδο Πολυδιαδρομικής Μεταφοράς.

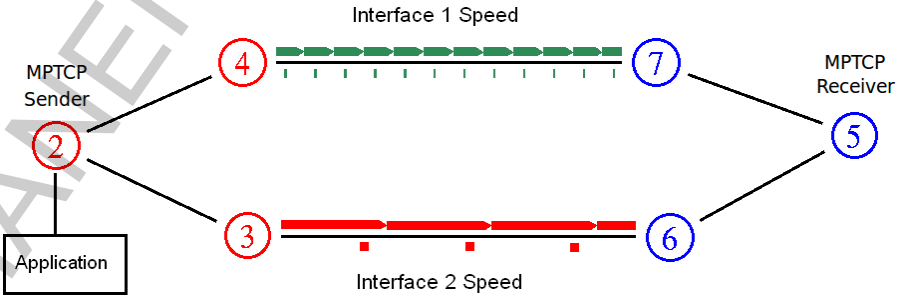
Πίνακας 2: Πίνακας συσχέτισης συμβάντων και ενεργειών που έχουν υλοποιηθεί στον Διαχειριστή Μονοπατιών

ΚΕΦΑΛΑΙΟ 4

ΑΞΙΟΛΟΓΗΣΗ ΤΟΥ MULTIPATH TCP ΣΕ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ ΕΠΙΚΟΙΝΩΝΙΑΣ

4.1 Προσομοίωσης πρωτοκόλλου στον ns2

Για την αξιολόγηση της απόδοσης του Multipath TCP, χρησιμοποιήθηκε επέκταση του προσομοιωτή ns2, που διατίθεται στην διεύθυνση http://code.google.com/p/multipath-tcp/. Σημειώνεται ότι υπάρχει η αντίστοιχη υλοποίηση και για τον προσομοιωτή ns3 που διατίθεται στην διεύθυνση http://code.google.com/p/mptcp-ns3/.



Σχήμα 14: Διάγραμμα ροής προσομοίωσης του MPTCP

Το σενάριο της προσομοίωσης περιλαμβάνει δύο κόμβους, οι οποίοι διαθέτουν δυο διεπαφές έκαστος. Οι δοκιμές περιλαμβάνουν δύο ειδών ζεύξεις, WiFi και 3G, σε συνδυασμό με την διαφοροποίηση δύο χαρακτηριστικών του δικτύου, του Λόγου Απώλειας Πακέτου (Packet Loss Ratio - PLR) και του Χρόνου Απόκρισης (Round Trip Time – RTT).

4.1.1 Script προσομοίωσης

Το παρακάτω script, σε γλώσσα προγραμματισμού TCL, χρησιμοποιήθηκε για την διενέργεια των προσομοιώσεων.

```
set ns [new Simulator]

#
# specify to print mptcp option information
#
Trace set show_tcphdr_ 2

#
# setup trace files
#
set f [open out.tr w]
$ns trace-all $f
set nf [open out.nam w]
$ns namtrace-all $nf


#
# mptcp sender
#
set n0 [$ns node]
set n0_0 [$ns node]
set n0_1 [$ns node]
```

```
$n0 color red
$n0_0 color red
$n0_1 color red
$ns multihome-add-interface $n0 $n0_0
$ns multihome-add-interface $n0 $n0_1

#
# mptcp receiver
#
set n1 [$ns node]
set n1_0 [$ns node]
set n1_1 [$ns node]
$n1 color blue
$n1_0 color blue
$n1_1 color blue
$ns multihome-add-interface $n1 $n1_0
$ns multihome-add-interface $n1 $n1_1

#
# normal tcp 1
#
set n2 [$ns node]
set n3 [$ns node]
$n2 color yellow
$n3 color yellow

#
# normal tcp 2
#
set n4 [$ns node]
set n5 [$ns node]
$n4 color green
$n5 color green

#
# intermediate nodes
#
set r1 [$ns node]
set r2 [$ns node]
set r3 [$ns node]
set r4 [$ns node]

$ns duplex-link $n0_0 $r1 10Mb 5ms DropTail
$ns duplex-link $r1 $r3   1Mb 5ms DropTail
```

```
$ns queue-limit $r1 $r3 30
$ns duplex-link $n1_0 $r3 10Mb 5ms DropTail

$ns duplex-link $n0_1 $r2 10Mb 5ms DropTail
$ns duplex-link $r2 $r4   1Mb 5ms DropTail
$ns queue-limit $r2 $r4 30
$ns duplex-link $n1_1 $r4 10Mb 5ms DropTail

$ns duplex-link $n2 $r1 10Mb 5ms DropTail
$ns duplex-link $r3 $n3 10Mb 5ms DropTail
$ns duplex-link $n4 $r2 10Mb 5ms DropTail
$ns duplex-link $r4 $n5 10Mb 5ms DropTail


#
# create mptcp sender
#
#     1. create subflows with
Agent/TCP/FullTcp/Sack/Multipath
#     2. attach subflow on each interface
#     3. create mptcp core
#     4. attach subflows to mptcp core
#     5. attach mptcp core to core node
#     6. attach application to mptcp core
#
set tcp0 [new Agent/TCP/FullTcp/Sack/Multipath]
$tcp0 set window_ 100
$ns attach-agent $n0_0 $tcp0
set tcp1 [new Agent/TCP/FullTcp/Sack/Multipath]
$tcp1 set window_ 100
$ns attach-agent $n0_1 $tcp1
set mptcp [new Agent/MPTCP]
$mptcp attach-tcp $tcp0
$mptcp attach-tcp $tcp1
$ns multihome-attach-agent $n0 $mptcp
set ftp [new Application/FTP]
$ftp attach-agent $mptcp


#
# create mptcp receiver
#
set mptcpsink [new Agent/MPTCP]
set sink0 [new Agent/TCP/FullTcp/Sack/Multipath]
```

```
$ns attach-agent $n1_0 $sink0
set sink1 [new Agent/TCP/FullTcp/Sack/Multipath]
$ns attach-agent $n1_1 $sink1
$mptcpsink attach-tcp $sink0
$mptcpsink attach-tcp $sink1
$ns multihome-attach-agent $n1 $mptcpsink
$ns multihome-connect $mptcp $mptcpsink
$mptcpsink listen


#
# create sack TCP connection
#
set reno0 [new Agent/TCP/FullTcp/Sack]
$reno0 set window_ 100
$ns attach-agent $n2 $reno0
set renosink0 [new Agent/TCP/FullTcp/Sack]
$ns attach-agent $n3 $renosink0
$ns connect $reno0 $renosink0
$renosink0 listen
set ftp0 [new Application/FTP]
$ftp0 attach-agent $reno0


#
# create sack TCP connection
#
set reno1 [new Agent/TCP/FullTcp/Sack]
$reno1 set window_ 100
$ns attach-agent $n4 $reno1
set renosink1 [new Agent/TCP/FullTcp/Sack]
$ns attach-agent $n5 $renosink1
$ns connect $reno1 $renosink1
$renosink1 listen
set ftp1 [new Application/FTP]
$ftp1 attach-agent $reno1

proc finish {} {
     global ns f
     global nf
     $ns flush-trace
     close $f
     close $nf
```

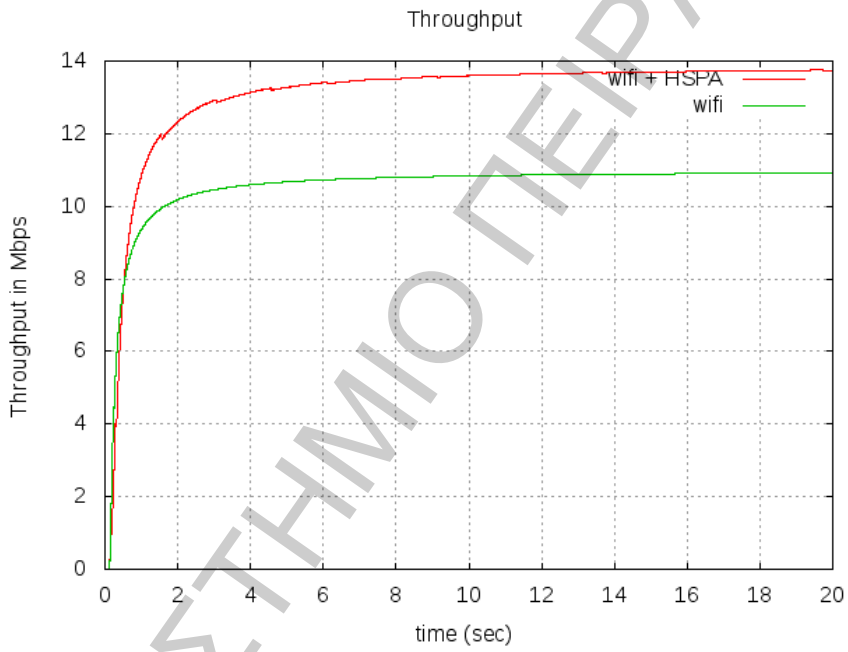
```
    set awkcode {
        {
            if ($1 == "r" && NF == 20) {
              if ($3 == "1" && $4 == "10" && $5 == "tcp")
{
                print $2, $18 >> "mptcp"
              }
              if ($3 == "2" && $4 == "11" && $5 == "tcp")
{
                print $2, $18 >> "mptcp"
              }
            }
            if ($1 == "r" && NF == 17) {
              if ($3 == "6" && $4 == "10" && $5 == "tcp")
{
                print $2, $11 >> "normal-tcp1"
              }
              if ($3 == "8" && $4 == "11" && $5 == "tcp")
{
                print $2, $11 >> "normal-tcp2"
              }
          }
        }
    }
     exec rm -f mptcp normal-tcp1 normal-tcp2
    exec awk $awkcode out.tr
    exec xgraph -M -m -nl mptcp normal-tcp1 normal-tcp2
     exit
}

$ns at 0.1 "$ftp start"
$ns at 0.1 "$ftp0 start"
$ns at 0.1 "$ftp1 start"
$ns at 300 "finish"
```

4.1.2 Αποτελέσματα προσομοίωσης του Multipath TCP

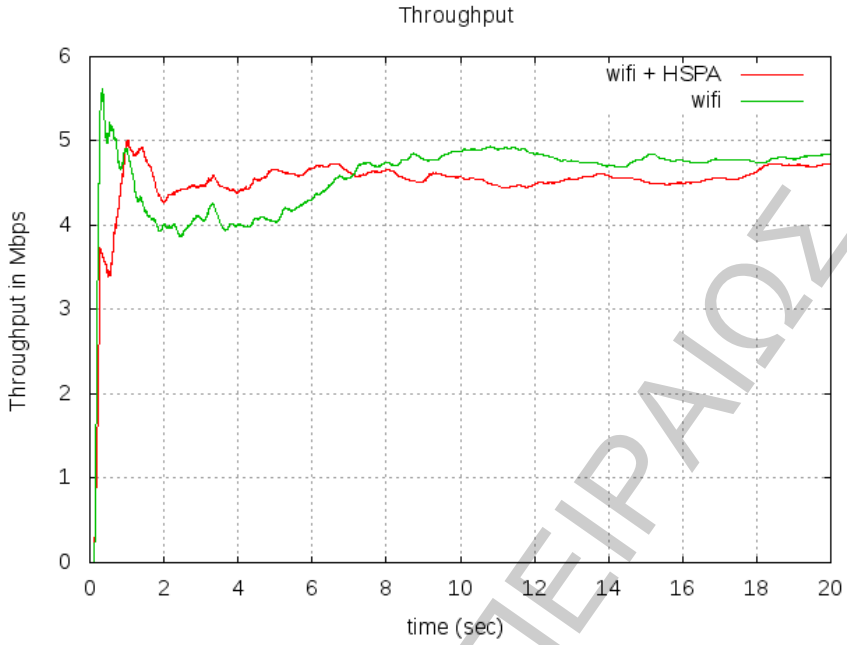
Στα επόμενα υποκεφάλαια παρουσιάζονται τα αποτελέσματα των προσομοιώσεων ανά σενάριο.

4.1.2.1 WiFi (11Mbps) + HSPA (3.2Mbps) με PLR = 0% και RTT = 10ms



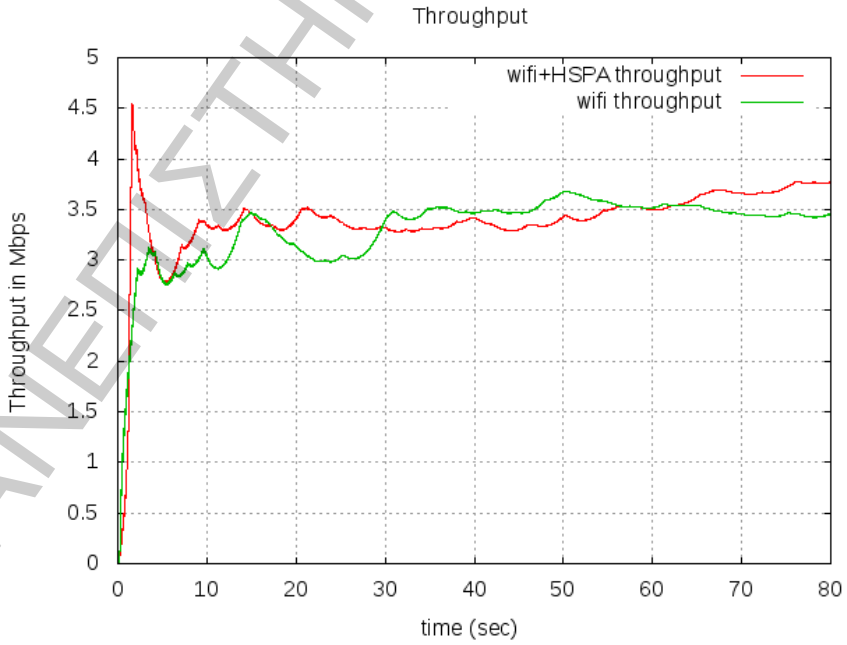
Σχήμα 15: Απόδοση MPTCP σε WiFi (11Mbps) + HSPA (3.2Mbps) με PLR = 0% και RTT = 10ms

4.1.2.2 WiFi (11Mbps) + HSPA (3.2Mbps) με PLR = 1% και RTT = 10ms



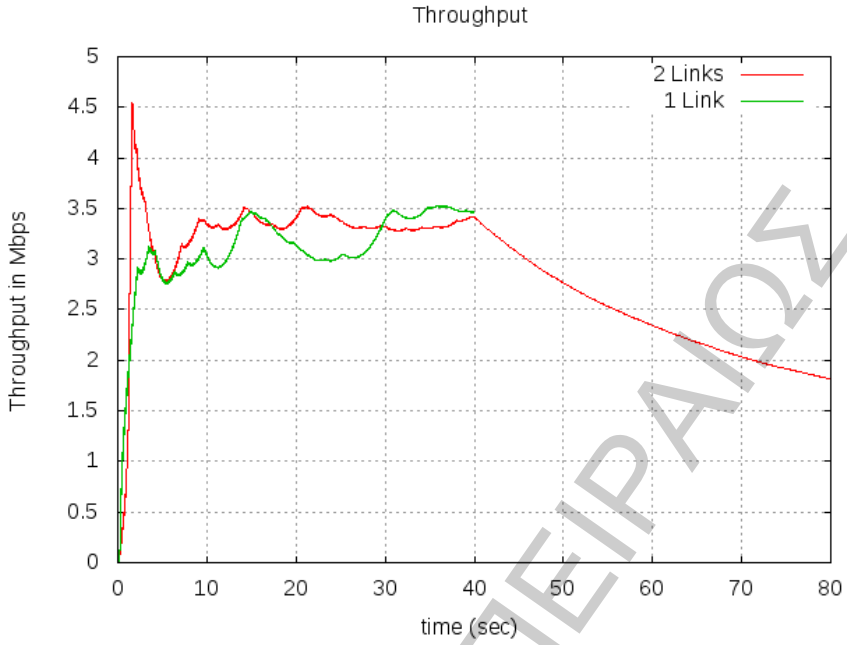
Σχήμα 16: Απόδοση MPTCP σε WiFi (11Mbps) + HSPA (3.2Mbps) με PLR = 1% και RTT = 10ms

4.1.2.3 WiFi (12Mbps) με PLR = 0.1%, RTT = 50ms και μικρό φορτίο + HSPA (5Mbps) με PLR = 0.5% και RTT = 120ms



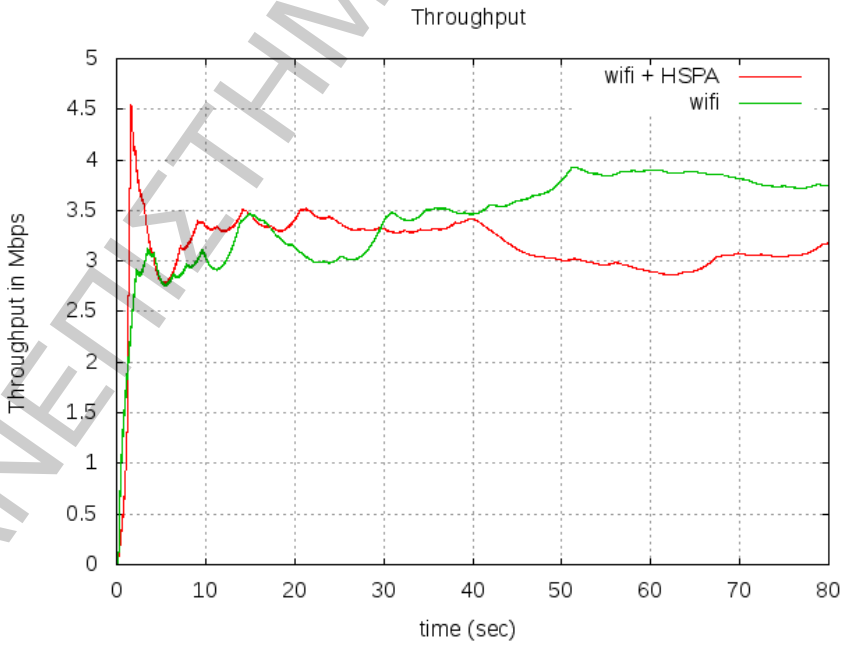
Σχήμα 17: Απόδοση MPTCP με PLR = 0.1%, RTT = 50ms και μικρό φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps)

4.1.2.4 Διακοπή της ζεύξης WiFi στην μέση της προσομοίωσης



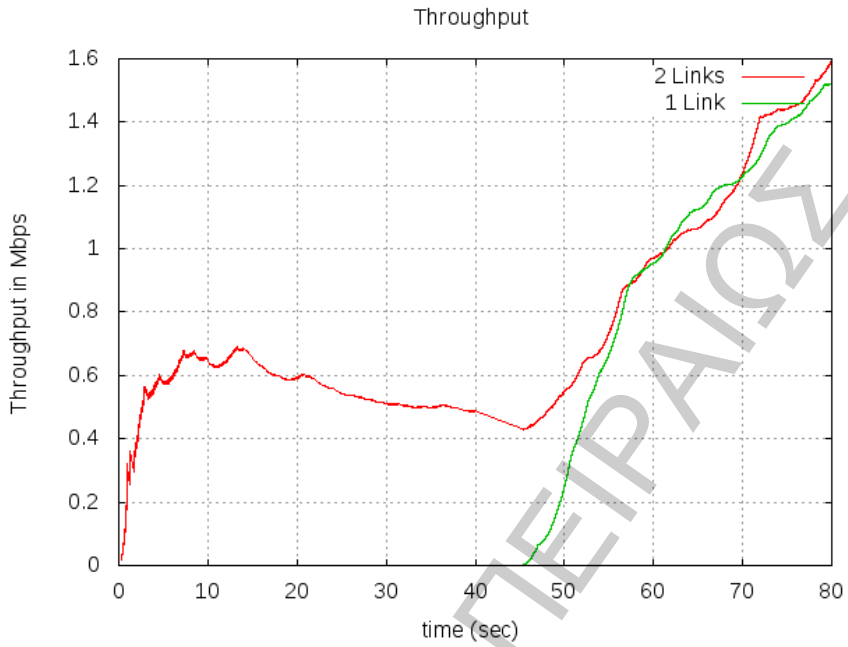
Σχήμα 18: Απόδοση MPTCP με PLR = 0.1%, RTT = 50ms και μικρό φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Διακοπή της ζεύξης WiFi στην μέση της προσομοίωσης

4.1.2.4 Διακοπή της ζεύξης HSPA στην μέση της προσομοίωσης



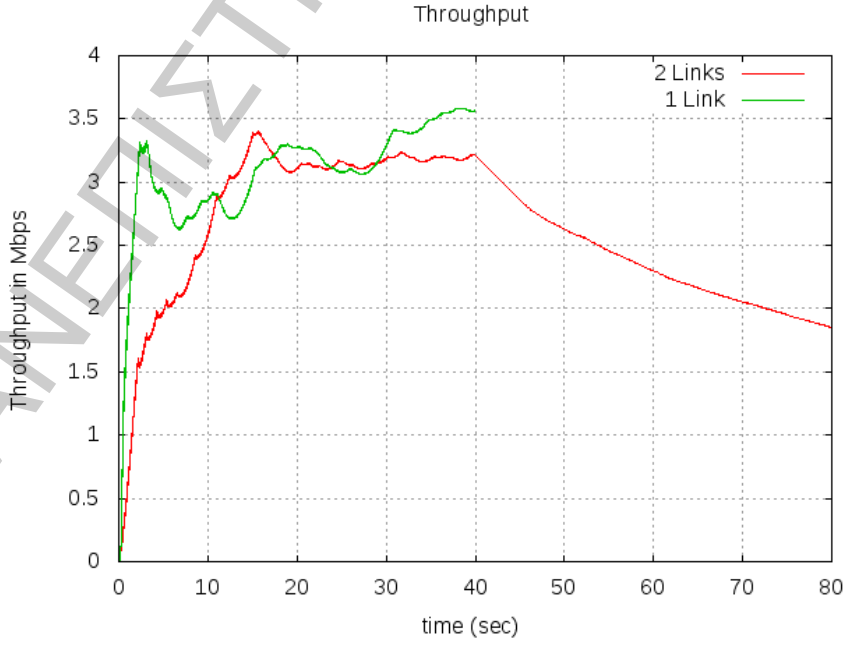
Σχήμα 19: Απόδοση MPTCP με PLR = 0.1%, RTT = 50ms και μικρό φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Διακοπή της ζεύξης HSPA στην μέση της προσομοίωσης

4.1.2.5 Εκκίνηση συνόδου με HSPA και αλλαγή της ενεργής ζεύξης από HSPA σε WiFi με μικρό φορτίο στην μέση της προσομοίωσης



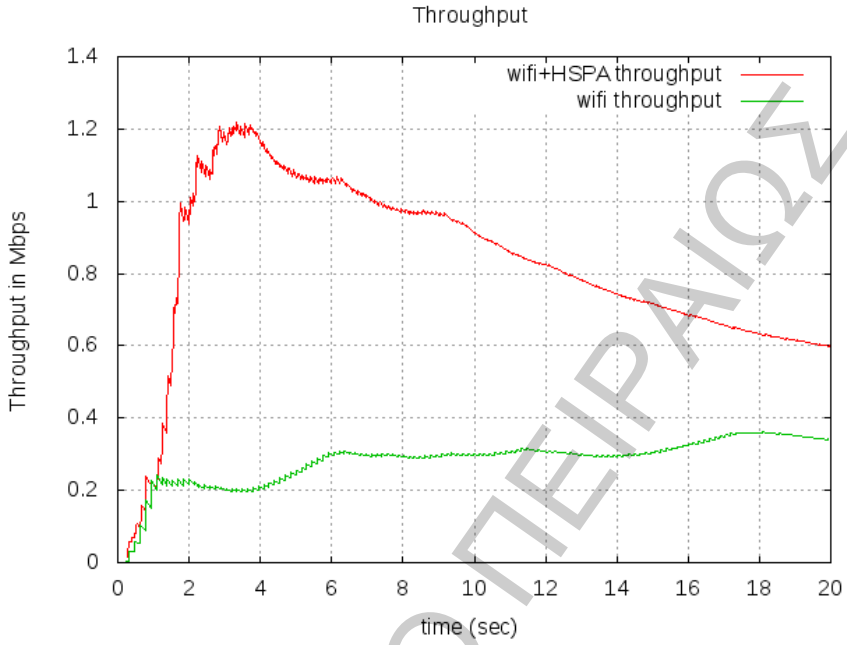
Σχήμα 20: Απόδοση MPTCP με PLR = 0.1%, RTT = 50ms και μικρό φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Εκκίνηση συνόδου με HSPA και αλλαγή της ενεργής ζεύξης από HSPA σε WiFi (με μικρό φορτίο) στην μέση της προσομοίωσης

4.1.2.6 Εκκίνηση συνόδου με WiFi (με μικρό φορτίο) και αλλαγή της ενεργής ζεύξης από WiFi σε HSPA στην μέση της προσομοίωσης



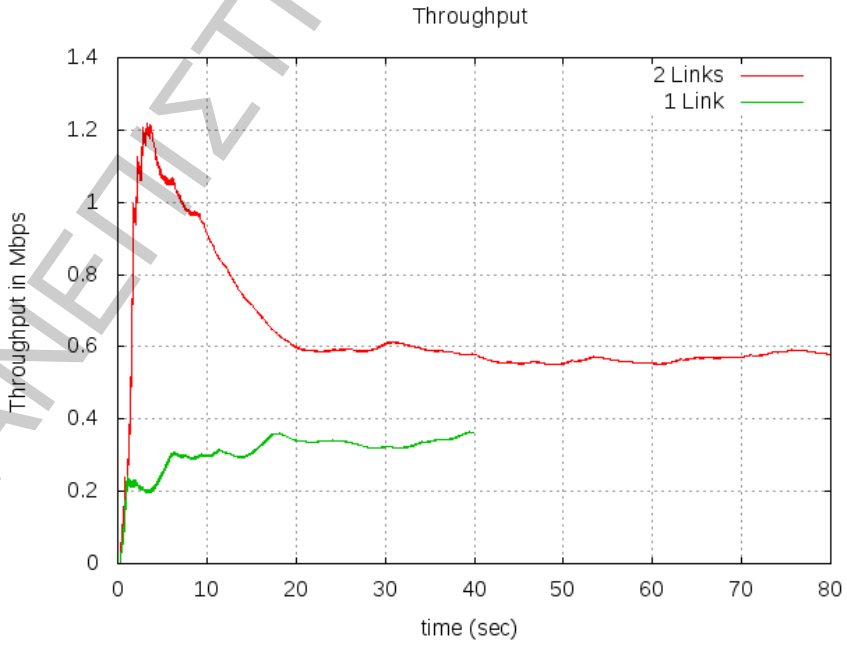
Σχήμα 21: Απόδοση MPTCP με PLR = 0.1%, RTT = 50ms και μικρό φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Εκκίνηση συνόδου με WiFi (με μικρό φορτίο) και αλλαγή της ενεργής ζεύξης από WiFi σε HSPA στην μέση της προσομοίωσης

4.1.2.7 WiFi (12Mbps) με PLR = 1%, RTT = 150ms και μεγάλο φορτίο + HSPA (5Mbps) με PLR = 0.5% και RTT = 120ms



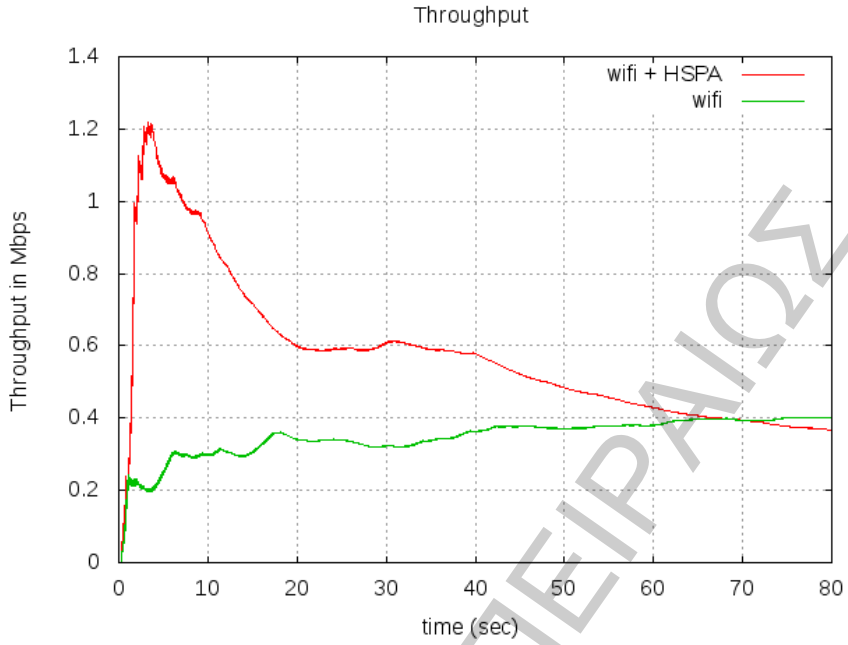
Σχήμα 22: Απόδοση MPTCP με PLR = 1%, RTT = 150ms και μεγάλο φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps)

4.1.2.8 Διακοπή της ζεύξης WiFi στην μέση της προσομοίωσης



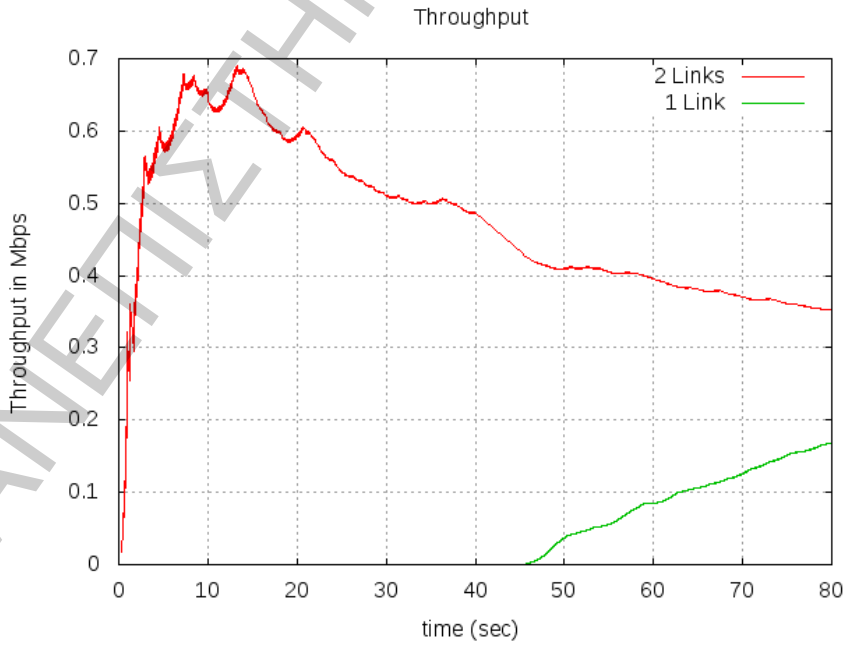
Σχήμα 23: Απόδοση MPTCP με PLR = 1%, RTT = 150ms και μεγάλο φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Διακοπή της ζεύξης WiFi στην μέση της προσομοίωσης

4.1.2.9 Διακοπή της ζεύξης HSPA στην μέση της προσομοίωσης



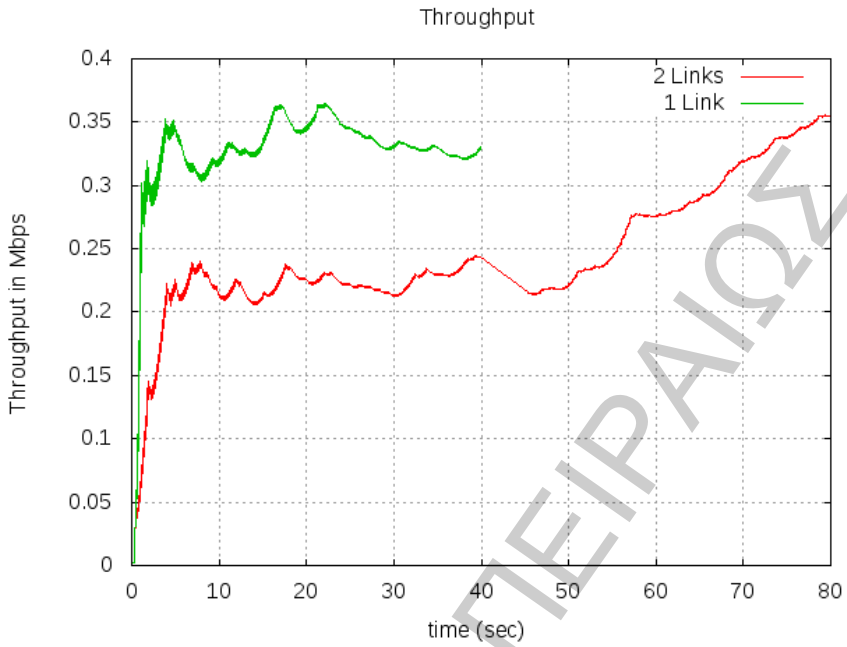
Σχήμα 24: Απόδοση MPTCP με PLR = 1%, RTT = 150ms και μεγάλο φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Διακοπή της ζεύξης HSPA στην μέση της προσομοίωσης

4.1.2.10 Εκκίνηση συνόδου με HSPA και αλλαγή της ενεργής ζεύξης από HSPA σε WiFi με μικρό φορτίο στην μέση της προσομοίωσης



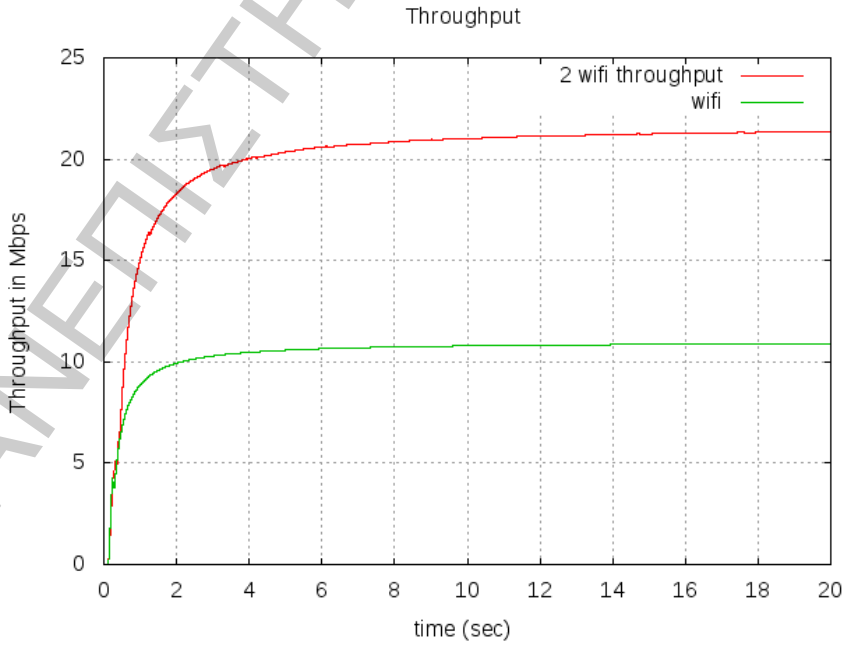
Σχήμα 25: Απόδοση MPTCP με PLR = 1%, RTT = 150ms και μεγάλο φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Εκκίνηση συνόδου με HSPA και αλλαγή της ενεργής ζεύξης από HSPA σε WiFi (με μικρό φορτίο) στην μέση της προσομοίωσης

4.1.2.11 Εκκίνηση συνόδου με WiFi (με μεγάλο φορτίο) και αλλαγή της ενεργής ζεύξης από WiFi σε HSPA στην μέση της προσομοίωσης



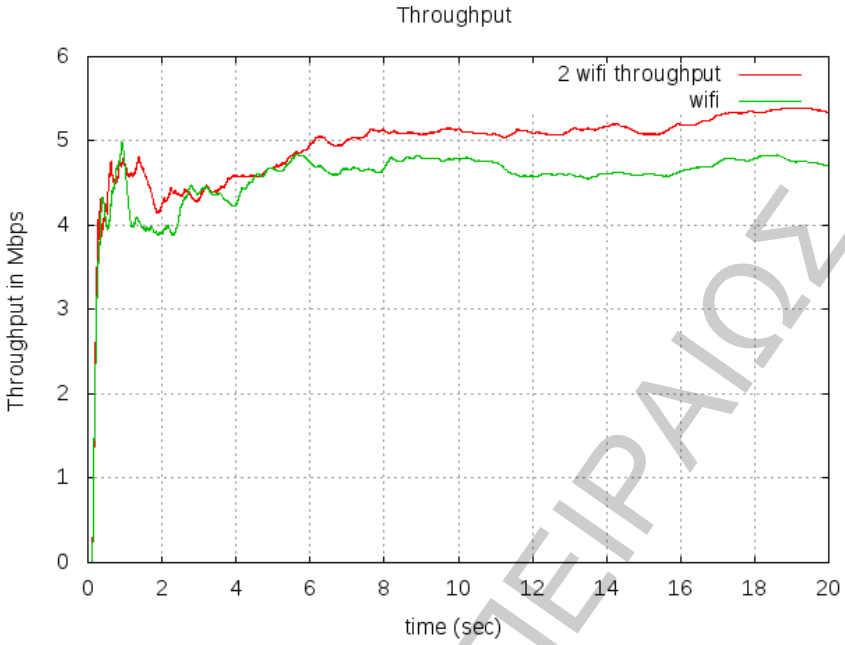
Σχήμα 26: Απόδοση MPTCP με PLR = 1%, RTT = 150ms και μεγάλο φορτίο στην ζεύξη WiFi (12Mbps), PLR = 0.5% και RTT = 120ms στην ζεύξη HSPA (5Mbps). Εκκίνηση συνόδου με WiFi (με μεγάλο φορτίο) και αλλαγή της ενεργής ζεύξης από WiFi σε HSPA στην μέση της προσομοίωσης

4.1.2.12 2 WiFi (11Mbps) με PLR = 0% και RTT = 10ms



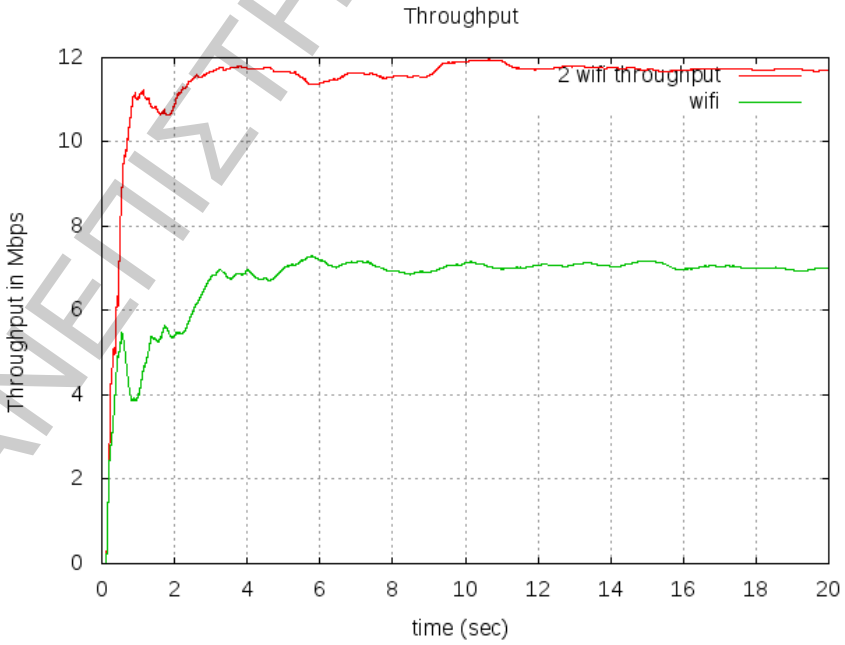
Σχήμα 27: Απόδοση MPTCP σε 2 WiFi (11Mbps) με PLR = 0% και RTT = 10ms

4.1.2.13 2 WiFi (11Mbps) με PLR = 1% και RTT = 10ms



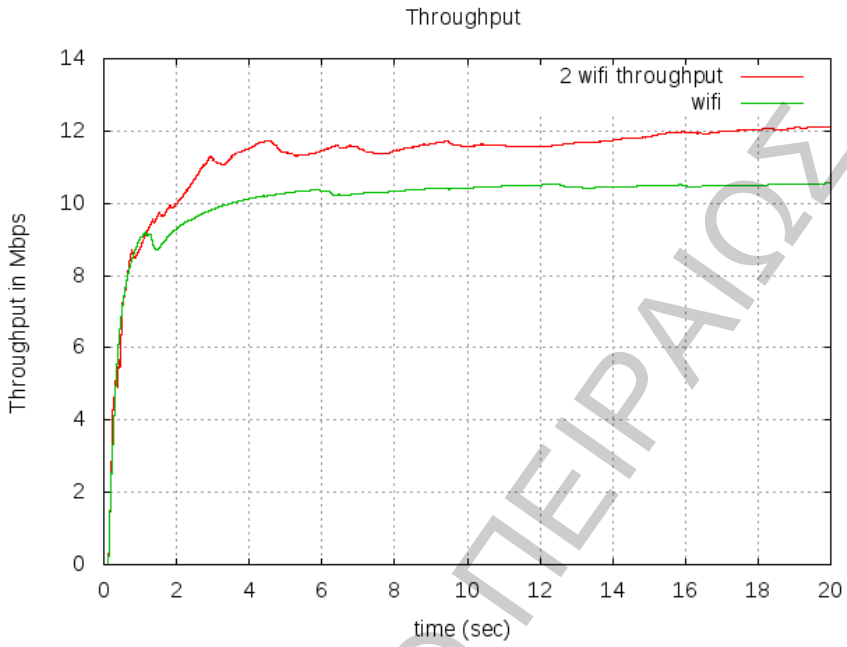
Σχήμα 28: Απόδοση MPTCP σε 2 WiFi (11Mbps) με PLR = 1% και RTT = 10ms

4.1.2.14 2 WiFi (11Mbps) με PLR = 0.1%, RTT = 10ms και PLR = 0.5%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 0.5% και RTT = 10ms



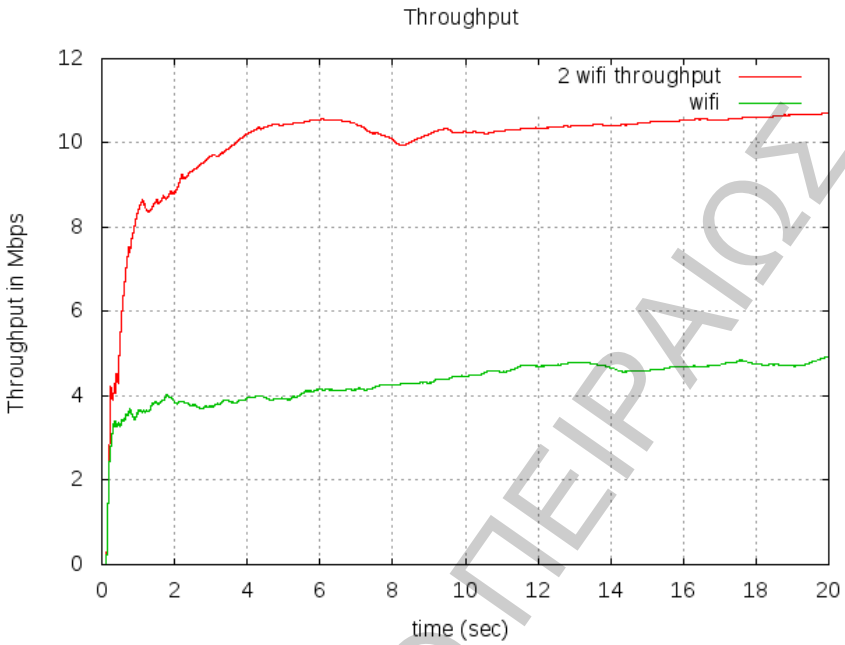
Σχήμα 29: Απόδοση MPTCP σε 2 WiFi (11Mbps) με PLR = 0.1%, RTT = 10ms και PLR = 0.5%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 0.5% και RTT = 10ms

4.1.2.15 2 WiFi (11Mbps) με PLR = 0.1%, RTT = 10ms και PLR = 0.5%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 0.1% και RTT = 10ms



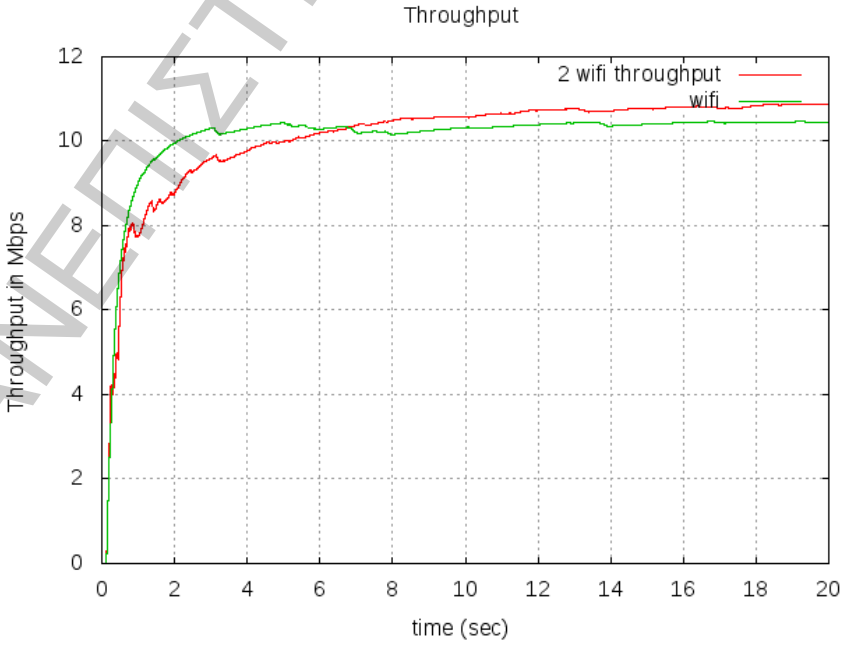
Σχήμα 30: Απόδοση MPTCP σε 2 WiFi (11Mbps) με PLR = 0.1%, RTT = 10ms και PLR = 0.5%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 0.1% και RTT = 10ms

4.1.2.16 2 WiFi (11Mbps) με PLR = 1%, RTT = 10ms και PLR = 0.1%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 1% και RTT = 10ms



Σχήμα 31: Απόδοση MPTCP σε 2 WiFi (11Mbps) με PLR = 1%, RTT = 10ms και PLR = 0.1%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 1% και RTT = 10ms

4.1.2.17 2 WiFi (11Mbps) με PLR = 1%, RTT = 10ms και PLR = 0.1%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 0.1% και RTT = 10ms



Σχήμα 32: Απόδοση MPTCP σε 2 WiFi (11Mbps) με PLR = 1%, RTT = 10ms και PLR = 0.1%, RTT = 10ms αντίστοιχα σε σύγκριση με την απόδοση TCP σε WiFi (11Mbps) με PLR = 0.1% και RTT = 10ms

4.2 Μετρήσεις απόδοσης του Multipath TCP

Για τις μετρήσεις απόδοσης του Multipath TCP, εγκαταστάθηκε η υλοποίηση του Multipath TCP σε laptop και server. Στη συνέχεια, διενεργήθηκαν δοκιμές με διαφοροποίηση του Λόγου Απώλειας Πακέτου (Packet Loss Ratio - PLR) και του Χρόνου Απόκρισης (Round Trip Time – RTT).

4.2.1 Εγκατάσταση MPTCP

Η έκδοση 0.6 της υλοποίησης του Multipath TCP, που χρησιμοποιήθηκε για τις δοκιμές, βασίζεται στην έκδοση 2.6.36 του Linux. Για τις ανάγκες δοκιμών χρησιμοποιήθηκε το λειτουργικό σύστημα Ubuntu Linux 11.10 (Oneiric Ocelot) 32-bit. Για την εγκατάσταση θα πρέπει να εκτελεστούν τα ακόλουθα βήματα:

1. Δημιουργία directory που θα περιέχει τον πηγαίο κώδικα του πυρήνα του Linux συμπεριλαμβανομένου και του MPTCP:
   - mkdir –p /usr/src/linux-mptcp
   - cd /usr/src/linux-mptcp

2. Ανάκτηση του πηγαίου κώδικα:
   - git clone -b mptcp_trunk git://scm.info.ucl.ac.be/mtcp.git
   - cd mtcp/

3. Αντιγραφή του τρέχοντος configuration file του συστήματος:

- cp /boot/config<SYSTEM_CONFIG_FILE> .config

4. Εφαρμογή του configuration file για την δημιουργία του νέου πυρήνα:
   - yes " | make oldconfig

5. Παραμετροποίηση διαδικασίας δημιουργίας του νέου πυρήνα:
   - make menuconfig

6. Στο μενού ρυθμίσεων θα πρέπει να επιλεγούν τα εξής:
   - **απενεργοποίηση** των TCP-SYN cookies (Networking support->Networking options->TCP/IP networking->IP: TCP syncookie support (SYN_COOKIES))
   - **απενεργοποίηση** του Net DMA (Device drivers->DMA Engine support->Network: TCP receive copy offload (NET_DMA))
   - **απενεργοποίηση** του MD5 signature (Networking support->Networking options->TCP/IP networking->TCP: MD5 Signature Option support (TCP_MD5SIG))
   - **ενεργοποίηση** του πρωτοκόλλου MPTCP (Networking support->Networking options->TCP/IP networking->MPTCP protocol (MPTCP)).
   - **ενεργοποίηση** του Policy-Routing (Networking support->Networking options->IP: advanced router->IP: policy routing (IP_MULTIPLE_TABLES))

7. Δημιουργία του νέου πυρήνα:
   - sed -rie 's/echo "\+"/#echo "\+"/' scripts/setlocalversion

- make-kpkg clean
- CONCURRENCY_LEVEL=`getconf _NPROCESSORS_ONLN` fakeroot make-kpkg --initrd --append-to-version=-mptcp kernel_image kernel_headers

8. Εγκατάσταση του νέου πυρήνα:
   - cd ..
   - dpkg -i linux-image-3.0.0-mptcp_3.0.0-mptcp-10.00.Custom_amd64.deb
   - dpkg -i linux-headers-3.0.0-mptcp_3.0.0-mptcp-10.00.Custom_amd64.deb

9. Επανεκκίνηση του υπολογιστή και φόρτωση του λειτουργικού με τον πυρήνα που δημιουργήθηκε. Η συγκεκριμένη επιλογή μπορεί να πραγματοποιηθεί κατά την εκκίνηση στο μενού του διαχειριστή εκκίνησης (grub). Εναλλακτικά μπορεί να τεθεί ως ο προεπιλεγμένος πυρήνας, τροποποιώντας κατάλληλα το configuration του διαχειριστή εκκίνησης.

10. Ρύθμιση των κανόνων δρομολόγησης προκειμένου να καταστεί δυνατή η πολυδιαδρομικότητα. Υποθέτωντας ότι το τερματικό διαθέτει δύο διεπαφές WiFi, οι απαραίτητες εντολές είναι:

    - ip rule add from 192.168.1.104 table 1
    - ip rule add from 192.168.2.4 table 2

    - ip route add 192.168.1.0/24 dev wlan0 scope link table 1

- ip route add default via 192.168.1.1 dev wlan0 table 1

- ip route add 192.168.2.0/24 dev wlan1 scope link table 2
- ip route add default via 192.168.2.1 dev wlan1 table 2

- ip route add default scope global nexthop via 192.168.1.1 dev wlan0

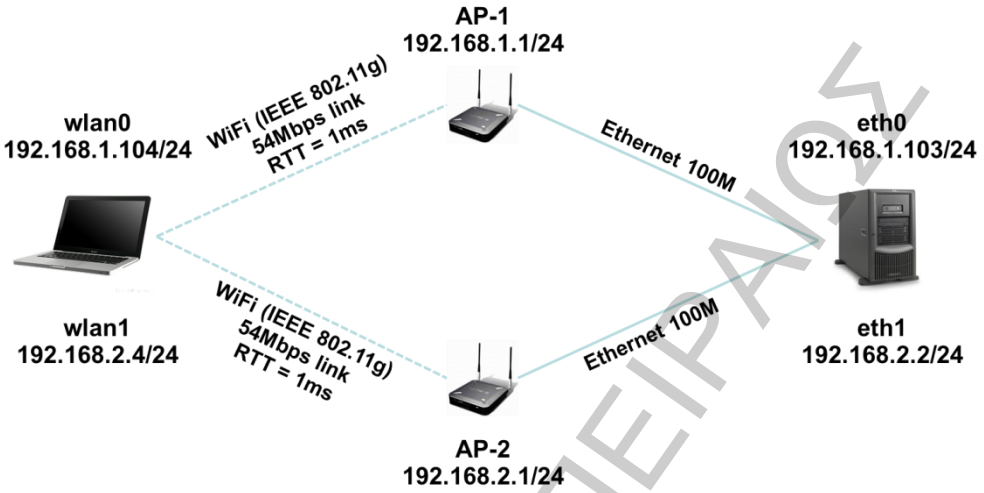
11. Για τις ανάγκες των δοκιμών, θα χρησιμοποιηθεί το εργαλείο netem, σε συνδυασμό με το tc (traffic control). Το netem παρέχει λειτουργία προσομοίωσης του δικτύου για την δοκιμή πρωτοκόλλων με την προσομοίωση των ιδιοτήτων των δικτύων ευρείας περιοχής.

- Για την προσθήκη επιπλέον απώλειας 0.5% (ακολουθώντας ομοιόμορφη κατανομή) στην διεπαφή wlan0:

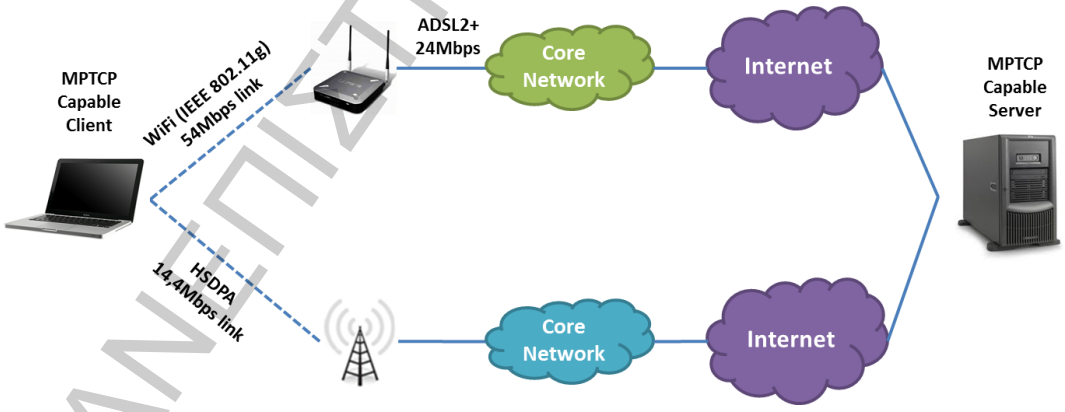
    - tc qdisc add dev wlan0 root netem loss 0.5%

- Για την προσθήκη επιπλέον χρονικής καθυστέρησης 100ms στην διεπαφή wlan0:

    - tc qdisc add dev wlan0 root netem delay 100ms



Σχήμα 33: Τοπολογία δικτύου δοκιμής με δύο διεπαφές WiFi

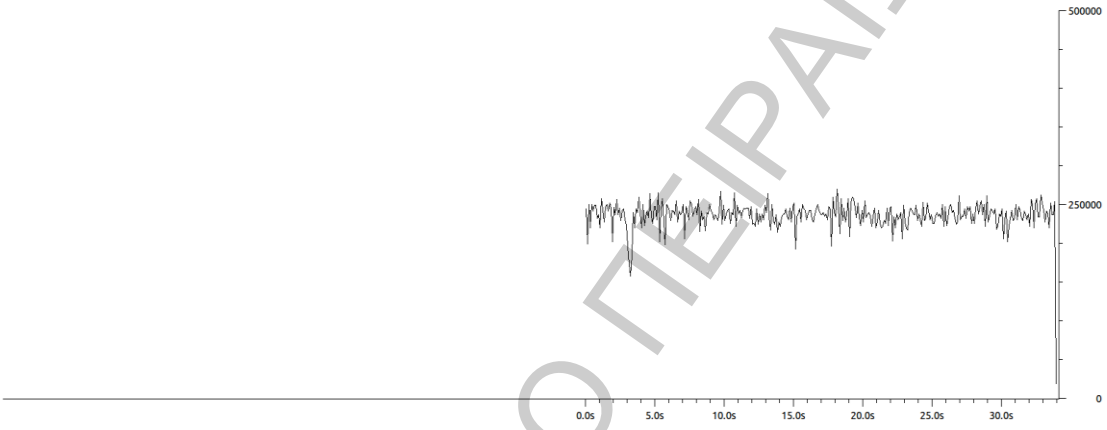


Σχήμα 34: Τοπολογία δικτύου δοκιμής με WiFi και 3G (HSDPA)

4.2.2 Αποτελέσματα δοκιμών με δύο διεπαφές WiFi

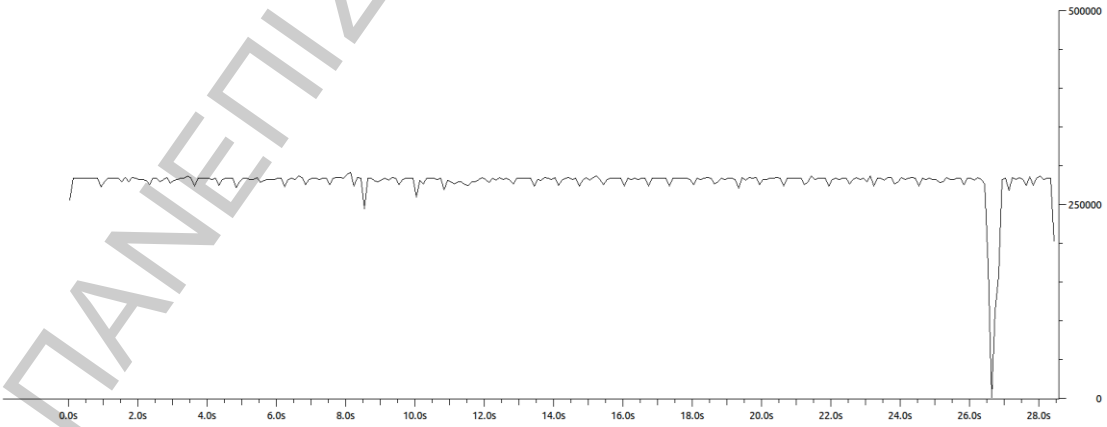
Στα επόμενα υποκεφάλαια παρουσιάζονται τα αποτελέσματα των δοκιμών ανά σενάριο.

4.2.2.1 TCP σε WiFi (διεπαφή wlan0)



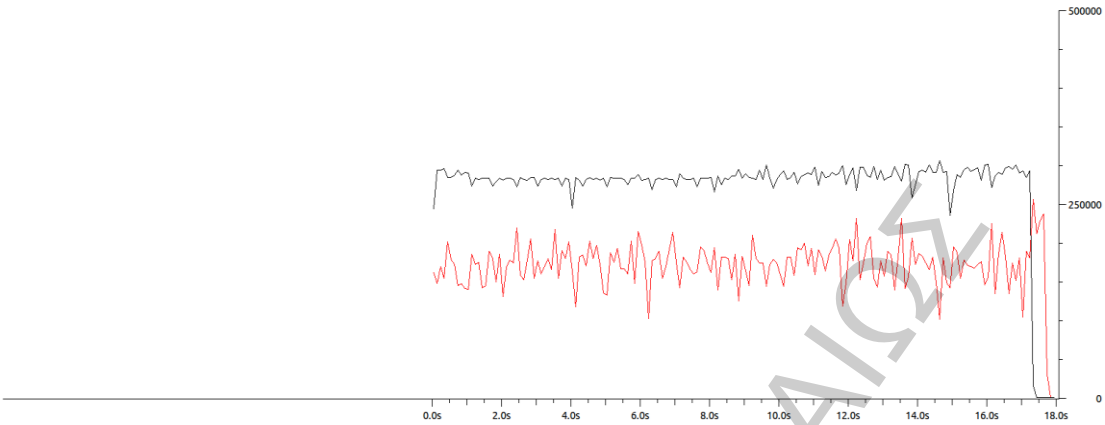
Σχήμα 35: Απόδοση TCP σε WiFi (διεπαφή wlan0)

4.2.2.2 TCP σε WiFi (διεπαφή wlan1)

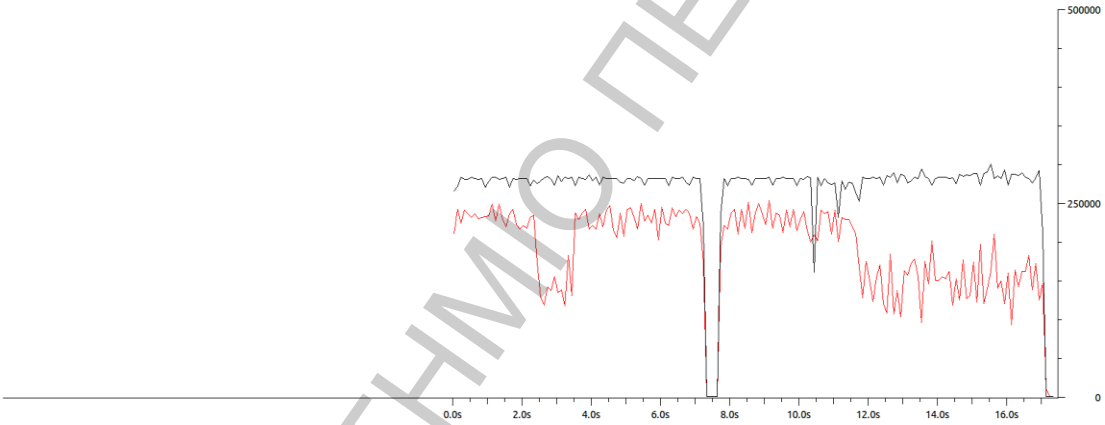


Σχήμα 36: Απόδοση TCP σε WiFi (διεπαφή wlan1)

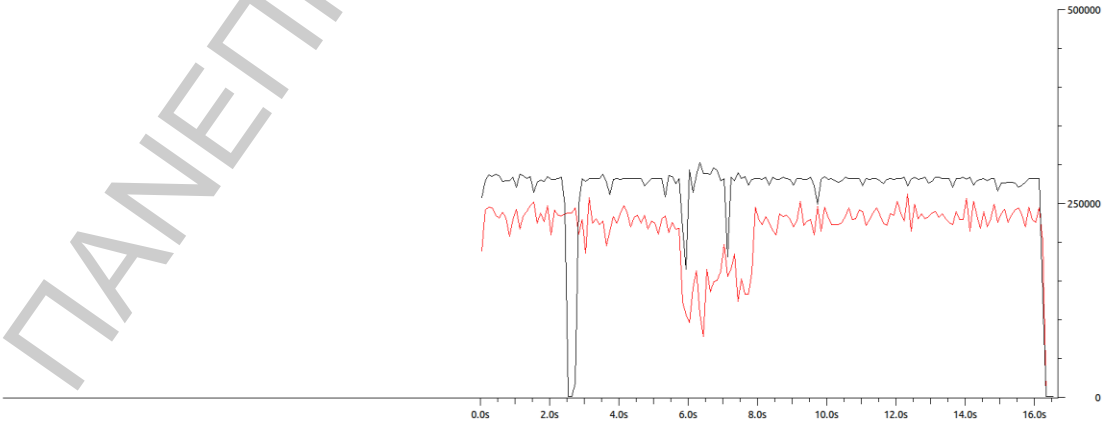
4.2.2.3 Multipath TCP σε δύο ενεργές διεπαφές WiFi



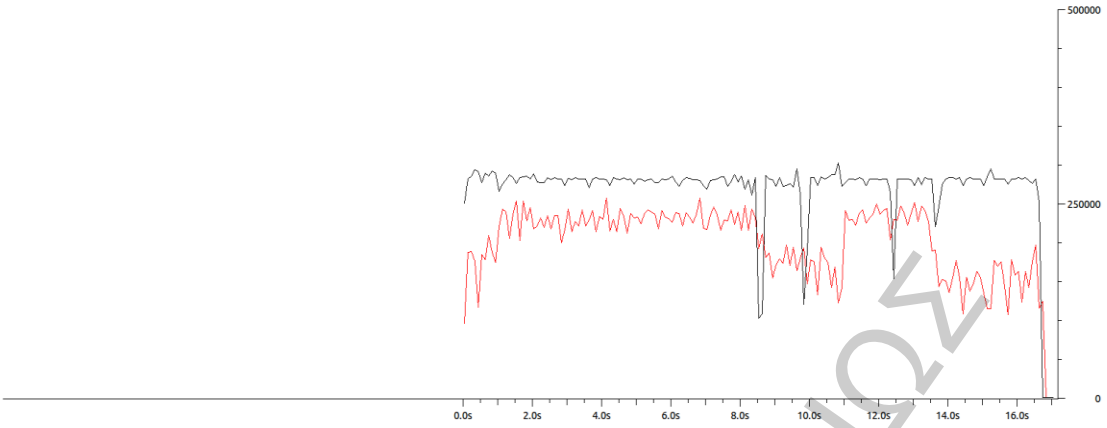
Σχήμα 37: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi (1)



Σχήμα 38: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi (2)

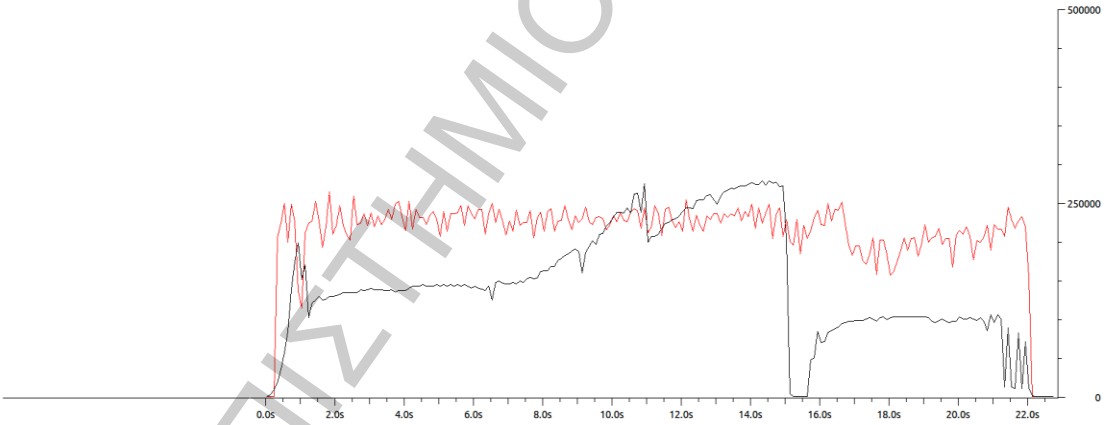


Σχήμα 39: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi (3)

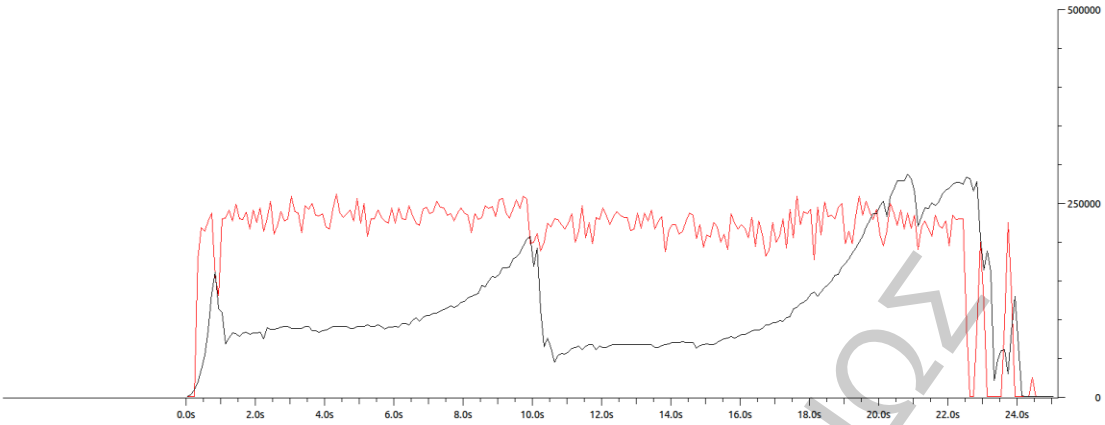


Σχήμα 40: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi (4)

4.2.2.4 Multipath TCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη καθυστέρηση 100ms στην διεπαφή wlan0

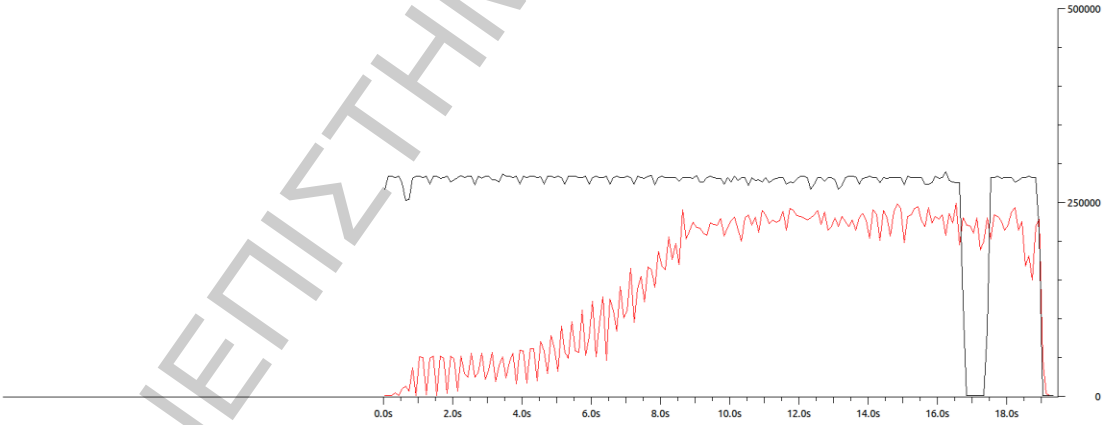


Σχήμα 41: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη καθυστέρηση 100ms στην διεπαφή wlan0 (μαύρη γραμμή) (1)

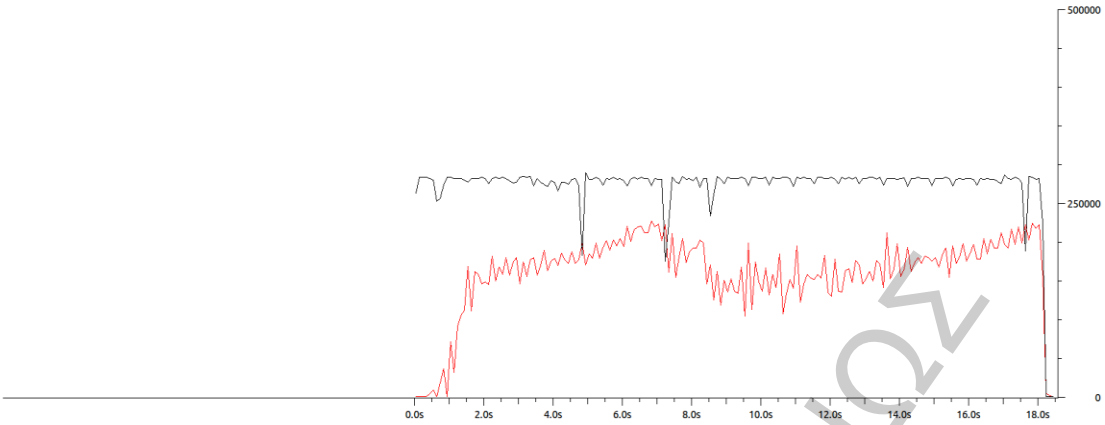


Σχήμα 42: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη καθυστέρηση 100ms στην διεπαφή wlan0 (μαύρη γραμμή) (2)

4.2.2.5 Multipath TCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη καθυστέρηση 150ms στην διεπαφή wlan1

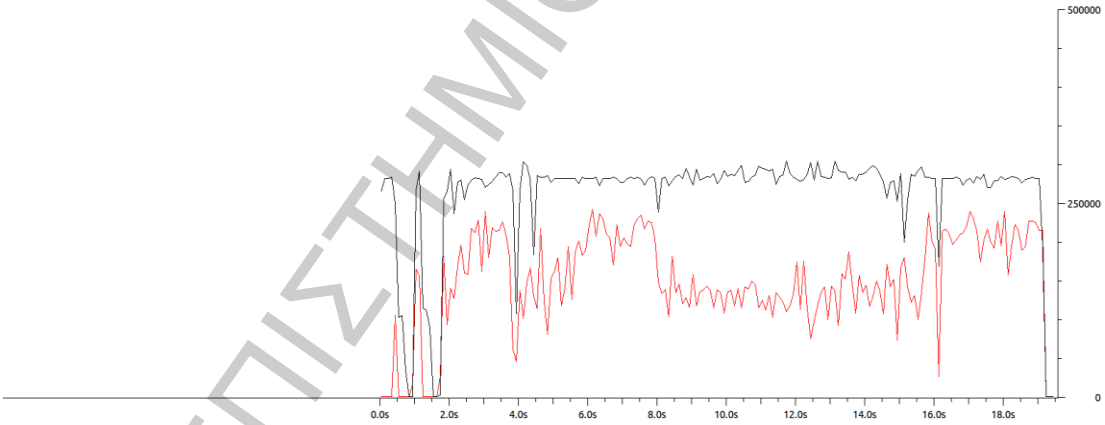


Σχήμα 43: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη καθυστέρηση 150ms στην διεπαφή wlan1 (κόκκινη γραμμή) (1)

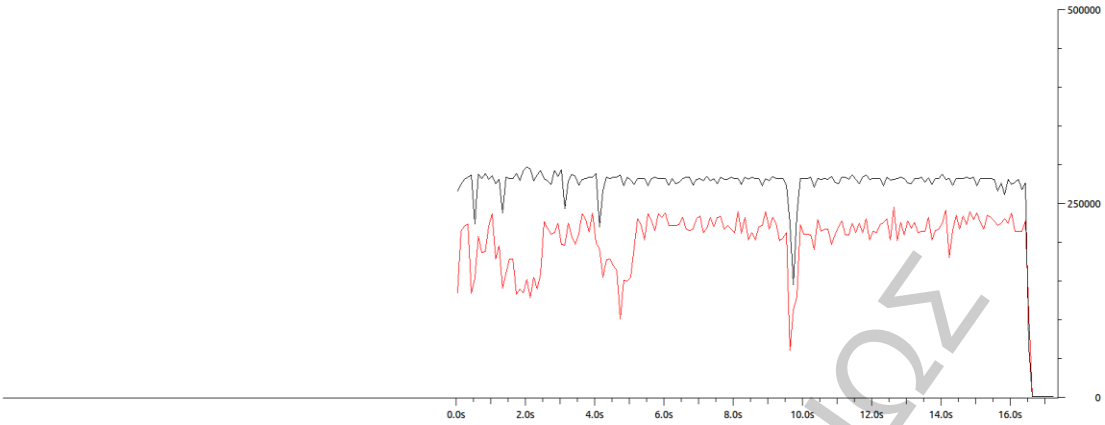


Σχήμα 44: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη καθυστέρηση 150ms στην διεπαφή wlan1 (κόκκινη γραμμή) (2)

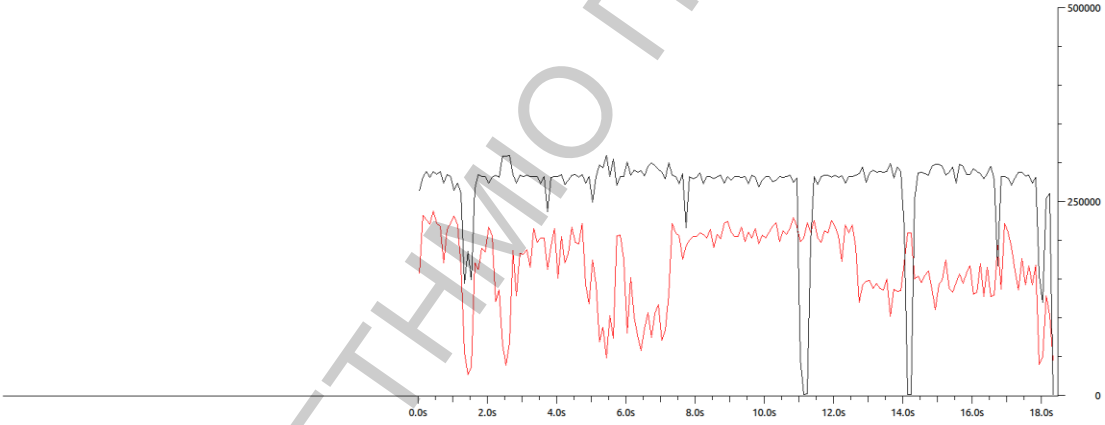
4.2.2.6 Multipath TCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.3% (ομοιόμορφη κατανομή) στην διεπαφή wlan0



Σχήμα 45: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.3% (ομοιόμορφη κατανομή) στην διεπαφή wlan0 (μαύρη γραμμή) (1)

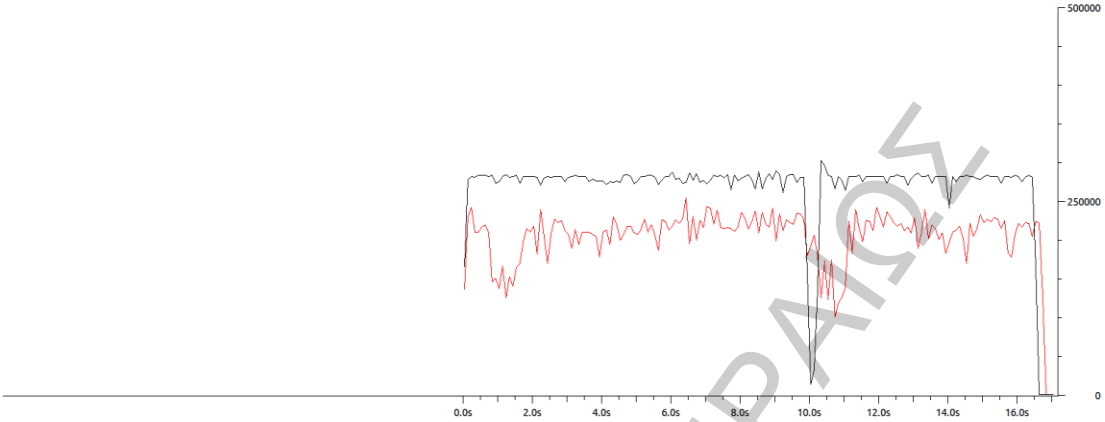


Σχήμα 46: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.3% (ομοιόμορφη κατανομή) στην διεπαφή wlan0 (μαύρη γραμμή) (2)

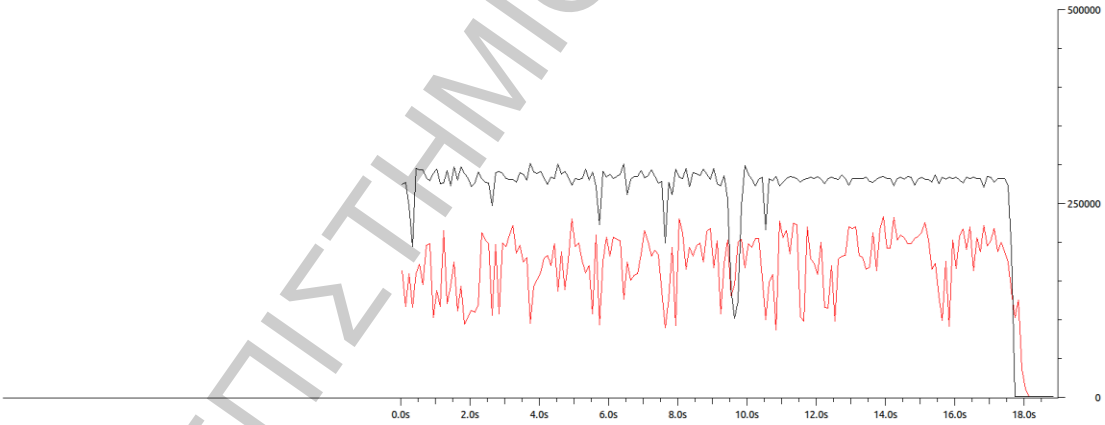


Σχήμα 47: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.3% (ομοιόμορφη κατανομή) στην διεπαφή wlan0 (μαύρη γραμμή) (3)

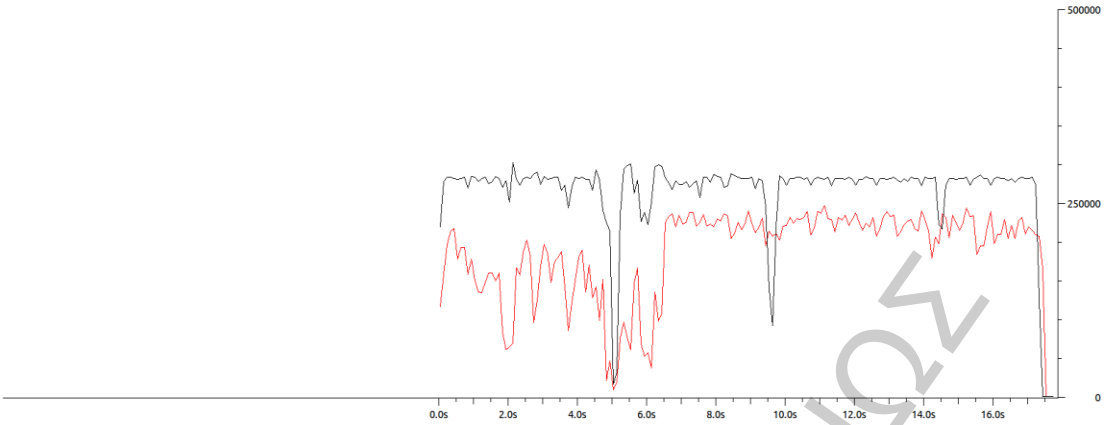
4.2.2.7 Multipath TCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.8% (ομοιόμορφη κατανομή) στην διεπαφή wlan1



Σχήμα 48: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.8% (ομοιόμορφη κατανομή) στην διεπαφή wlan1 (κόκκινη γραμμή) (1)

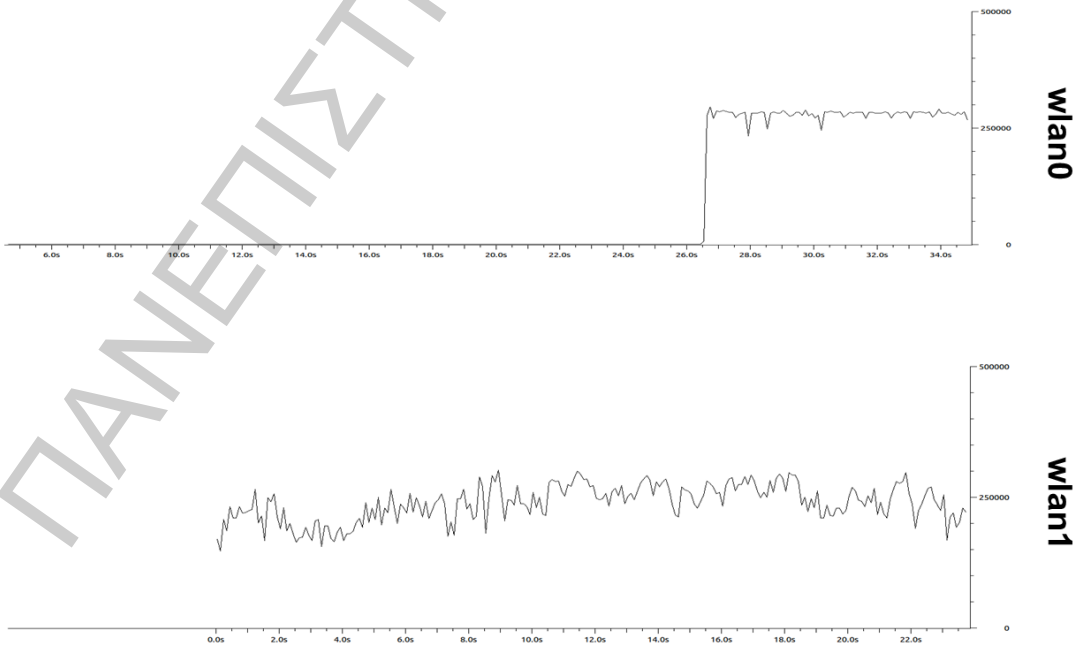


Σχήμα 49: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.8% (ομοιόμορφη κατανομή) στην διεπαφή wlan1 (κόκκινη γραμμή) (2)



Σχήμα 50: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με επιπρόσθετη απώλεια πακέτων 0.8% (ομοιόμορφη κατανομή) στην διεπαφή wlan1 (κόκκινη γραμμή) (3)

4.2.2.8 Multipath TCP σε δύο ενεργές διεπαφές WiFi με απώλεια διεπαφής wlan0 κατά τη διάρκεια της συνόδου



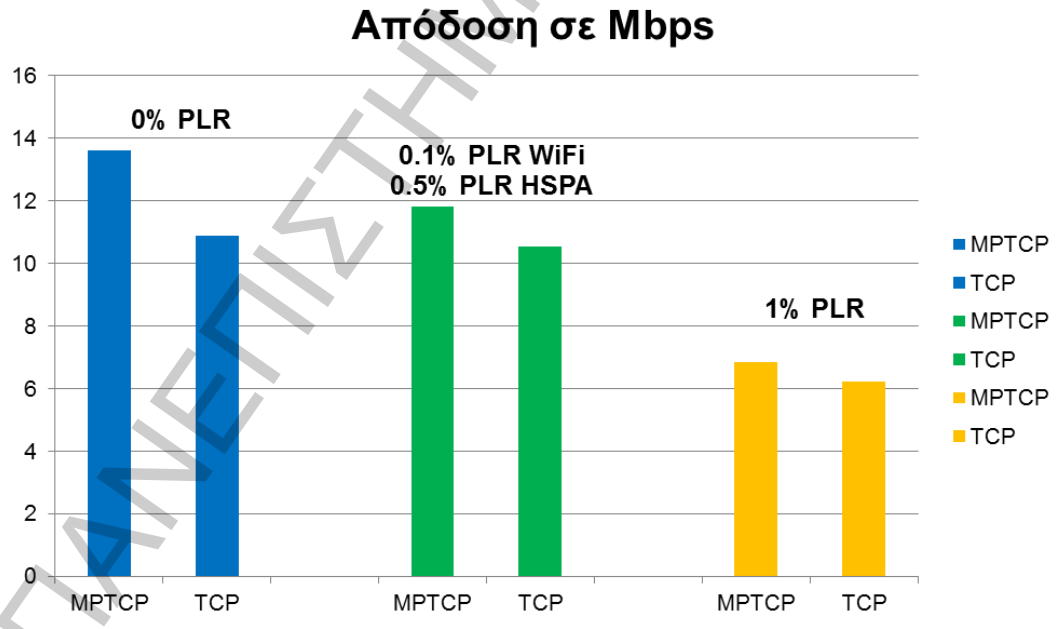
Σχήμα 51: Απόδοση MPTCP σε δύο ενεργές διεπαφές WiFi με απώλεια διεπαφής wlan0 κατά τη διάρκεια της συνόδου

4.3 Αποτελέσματα δοκιμών με WiFi και 3G (HSDPA)

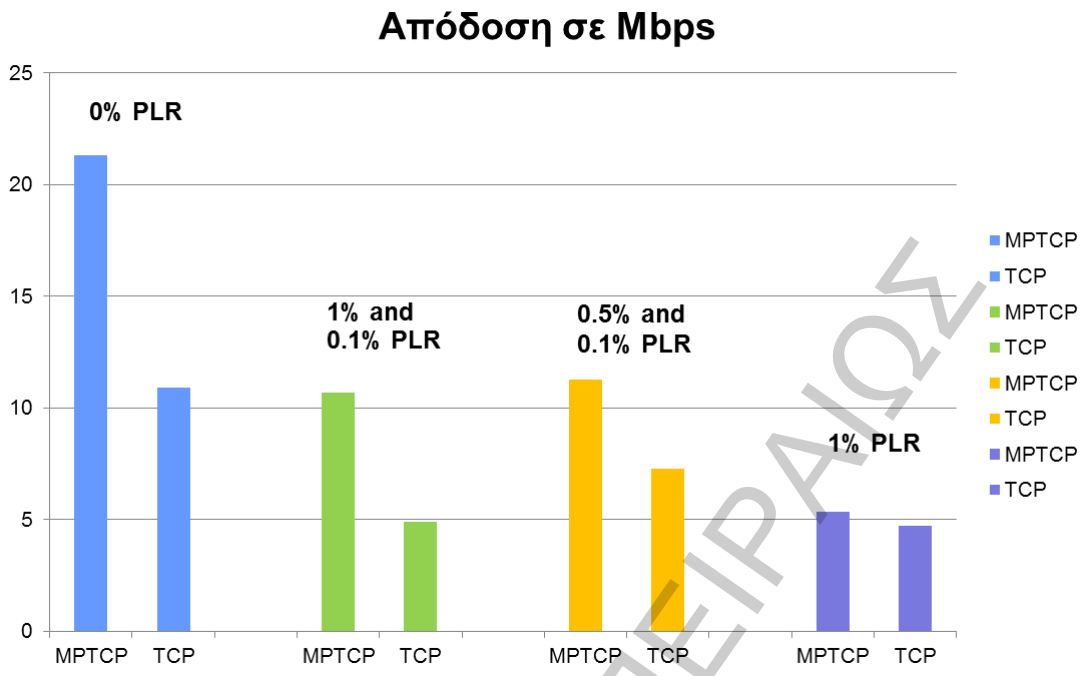
Παρά την προσπάθεια δοκιμής σε δύο παρόχους κινητής τηλεφωνίας (Cosmote και Vodafone), εντούτοις δεν κατέστει δυνατή η δοκιμή του Multipath TCP. Αυτό οφείλεται στο γεγονός ότι τα δίκτυα τρίτης γενιάς τροποποιούν το πακέτο σε επίπεδο μεταφοράς, με αποτέλεσμα να αφαιρούνται τα TCP Options που ενημερώνουν την άλλη πλευρά για την δυνατότητα υποστήριξης του πρωτοκόλλου. Έρευνα των [HNRG+11] δείχνει πως, σε ποσοστό 25% των συνολικών δειγμάτων, το πακέτο είχε τροποποιηθεί από ενδιάμεσους κόμβους (middleboxes).

ΣΥΜΠΕΡΑΣΜΑΤΑ

Από τις προσομοιώσεις, που πραγματοποιήθηκαν, προκύπτει το συμπέρασμα ότι το πρωτόκολλο Multipath TCP καταγράφει υψηλές επιδόσεις σε δίκτυα στα οποία επικρατούν καλές συνθήκες (ελάχιστο έως μηδενικό PLR και χαμηλό RTT). Ωστόσο, οι επιδόσεις φαίνονται να μειώνονται, όσο δυσχεραίνουν οι συνθήκες του δικτύου, πράγμα το οποίο είναι αναμενόμενο.



Σχήμα 52: Συγκεντρωτικός πίνακας αποτελεσμάτων απόδοσης TCP (μόνο WiFi) και Multipath TCP (WiFi + HSPA) στον προσομοιωτή NS2

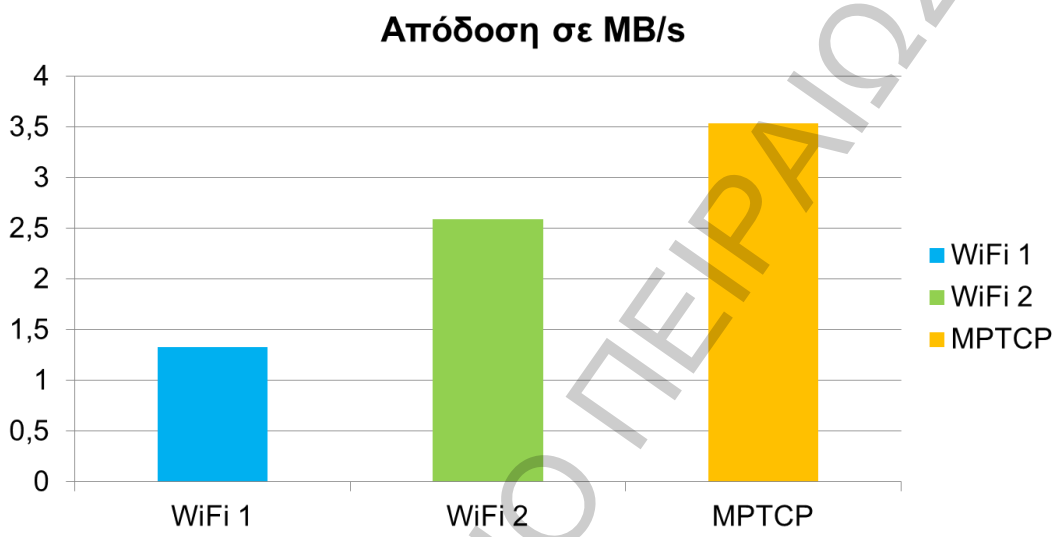


Σχήμα 53: Συγκεντρωτικός πίνακας αποτελεσμάτων απόδοσης TCP (1 WiFi) και Multipath TCP (2 WiFi) στον προσομοιωτή NS2

Αυτό που επίσης παρατηρείται, τόσο στην προσομοίωση όσο και στις δοκιμές, είναι το γεγονός ότι η υλοποίηση του Προγραμματιστή (scheduler) δεν πληροί τους στόχους που έχουν τεθεί στον αλγόριθμο ελέγχου συμφόρησης (§2.4). Αυτό το γεγονός, ενδεχομένως, να οδηγήσει σε μη-δίκαιη κατανομή της χωρητικότητας ανάμεσα στις συνδέσεις, με το Multipath TCP να λαμβάνει μεγαλύτερο μερίδιο απ' όσο θα έπρεπε.

Το μεγαλύτερο ενδεχομένως πρόβλημα που αντιμετωπίζει ένα πολυδιαδρομικό πρωτόκολλο μεταφοράς είναι η ύπαρξη των ενδιάμεσων κόμβων (middleboxes). Στις δοκιμές που διενεργήθηκαν, υπήρξε πλήρης αποτυχία στην απόπειρα προσπάθειας χρήσης του πρωτοκόλλου Multipath TCP μέσω των δικτύων 3G (HSDPA). Αυτό επιβεβαιώνεται και από την έρευνα των [HNRG+11] που δείχνει πως, σε ποσοστό

25% των συνολικών δειγμάτων, το πακέτο είχε τροποποιηθεί από ενδιάμεσους κόμβους. Ενδεχομένως το πρόβλημα να εξαλειφτεί μελλοντικά, όταν οι κωδικοί των TCP Options που χρησιμοποιούνται από το πρωτόκολλο να γνωστοποιηθούν και να ενσωματωθούν στις νέες υλοποιήσεις των κορμών δικτύων των παρόχων.



Σχήμα 54: Συγκεντρωτικός πίνακας αποτελεσμάτων μετρήσεων απόδοσης με 2 διεπαφές WiFi

Η εξοικονόμηση ενέργειας στα κινητά τερματικά ήταν ανέκαθεν ένα σημαντικό ζήτημα. Η ταυτόχρονη χρήση δύο διεπαφών θα πρέπει να πραγματοποιείται μόνο στην περίπτωση όπου υπάρχει ζήτηση μεγαλύτερου ρυθμού μεταγωγής δεδομένων από τον προσφερόμενο. Επομένως, θα πρέπει να σχεδιαστεί ένας Προγραμματιστής (scheduler) για την διανομή της κίνησης στις εκάστοτε διεπαφές, που θα έχει ως κριτήρια, εκτός των άλλων, το ενεργειακό κόστος της χρήσης κάθε επαφής, καθώς επίσης και το οικονομικό κόστος.

Στα σύγχρονα «έξυπνα τηλέφωνα» (smartphones), η διεπαφή 3G/4G παύει πλέον να είναι λειτουργική όταν εντοπιστεί ένα γνωστό (στο κινητό) δίκτυο WiFi, με αποτέλεσμα να δρομολογούνται όλα τα πακέτα μέσω της διεπαφής WiFi, ακόμα και

αν δεν υπάρχει συνδεσιμότητα στο Διαδίκτυο. Συνεπώς, απαιτούνται αλλαγές στα λειτουργικά συστήματα των κινητών τηλεφώνων, ούτως ώστε να υπάρχει η δυνατότητα ταυτόχρονης χρήσης των δύο διεπαφών. Επίσης, θα πρέπει να ρυθμίζεται αυτόματα ο πίνακας δρομολόγησης, πράγμα που μέχρι τώρα πραγματοποιείται χειροκίνητα.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[ALD+05] J. Abley, K. Lindqvist, E. Davies, B. Black, and V.Gill. IPv4 Multihoming Practices and Limitations. RFC 4116, Ιούλιος 2005.

[ASL04] A. Abd El Al, T. N. Saadawi, and M. J. Lee. LS-SCTP: a bandwidth aggregation technique for stream control transmission protocol. *Comput. Commun*., 27(10):1012-1024, Ιούνιος 2004.

[APS99] M. Allman, V. Paxson and W. Stevens. TCP Congestion Control. RFC 2581, Απρίλιος 1999.

[Aug10] B. Augustin. Tracing Internet Routes under Load Balancing. PhD thesis, Οκτώβριος 2010.

[Bag11] M. Bagnulo. Threat Analysis for TCP Extensions for Multipath Operation with Multiple Addresses. IETF RFC 6181, Μάρτιος 2011.

[Bar11] Sébastien Barré, Implementation and Assessment of Modern Host-based Multipath Solutions, PhD thesis, Université Catholique de Louvain (Belgium), Οκτώβριος 2011.

[BKG+01] J. Border, M. Kojo, J. Griner, G. Montenegro, and Z. Shelby. Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations. RFC 3135, Ιούνιος 2011.

[BOP94] L.S. Brakmo, S.W. O'Malley, and L.L. Peterson. TCP Vegas: New techniques for congestion detection and avoidance, volume 24. ACM, Οκτώβριος 1994.

[deL05] C. de Launois. Unleashing Traffic Engineering for IPv6 Multihomed Sites. PhD thesis. Université Catholique de Louvain (Belgium), Σεπτέμβριος 2005.

[EH06] D. Eastlake 3rd and T. Hansen. US Secure Hash Algorithms (SHA and HMAC-SHA). RFC 4634, Ιούλιος 2006. Αντικαταστάθηκε από το RFC 6234.

[FPKS+05] R. Fonseca, G. Porter, R. Katz, S. Shenker, and I. Stoica. IP options are not an option. Tech. Rep. UCB/EECS- 2005-24, 2005.

[FRH+11] A. Ford, C. Raiciu, M. Handley, S. Barre and J. Iyengar. Architectural Guidelines for Multipath TCP Development. IETF RFC 6182, Μάρτιος 2011.

[FRHB12] A. Ford, C. Raiciu, M. Handley and O. Bonaventure. TCP Extensions for Multipath Operation with Multiple Addresses. Internet draft, draft-ietf-mptcp-multiaddressed-08.txt, Μάιος 2012.

[Gao01] Lixin Gao. On Inferring Autonomous System Relationships in the Internet. *IEEE/ACM Transactions on Networking (TON)*, 9(6), Δεκέμβριος 2001

[HNRG+11] M. Honda, Y. Nishida, C. Raiciu, A. Greenhalgh, M. Handley, and H. Tokuda. Is it Still Possible to Extend TCP?. *IMC'11*, Νοέμβριος 2011

[HRX08] S. Ha, I. Rhee, and L. Xu. CUBIC: A new TCP-friendly high-speed TCP variant. *ACM SIGOPS Operating Systems Review*, 42(5):64–74, Ιούλιος 2008.

[HS02] H.-Y. Hsieh and R. Sivakumar. pTCP: An End-to-End Transport Layer Protocol for Striped Connections. In *International Conference on Network Protocols (ICNP)*, 10th, σελίδες 24–33. IEEE, Νοέμβριος 2002.

[IAS06] J. Iyengar, P.D. Amer, and R.R. Stewart. Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths. *IEEE/ACM Transactions on Networking (TON)*, 14(5):951–964, Οκτώβριος 2006.

[LBS08] S. Liu, T. Basar, and R. Srikant. TCP-Illinois: A loss-and delaybased congestion control algorithm for high-speed networks. *Performance Evaluation*, 65(6-7):417–440, Ιούνιος 2008.

[LIJM+10] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. *ACM SIGCOMM Computer Communications Review (CCR)*, 40(4):75–86, Οκτώβριος 2010.

[LWZ08] J. Liao, J. Wang, and X. Zhu. cmpSCTP: An extension of SCTP to support concurrent multi-path transfer. *International Conference of Communications (ICC)*, σελίδες 5762-5766. IEEE, Μάιος 2008.

[MK01] L. Magalhaes and R. Kravets. Transport Level Mechanisms for Bandwidth Aggregation on Mobile Hosts. In *International Conference on Network Protocols (ICNP)*, 9th, σελίδες 165–171. IEEE Computer Society, Νοέμβριος 2001.

[Nag84] J. Nagle. Congestion Control in IP/TCP Internetworks. RFC 896, Ιανουάριος 1984.

[NRO10] NRO (Number Resource Organization). Remaining IPv4 Address Space Drops Below 5%. http://www.nro.net/news/remaining-ipv4-address-space-dropsbelow-5, Οκτώβριος 2010. [Ανακτήθηκε στις 03-04-2012].

[Ong09] A. Ongenae. Multi-path congestion control. Master's thesis, Université Catholique de Louvain (Belgium), 2009.

[Pos80] J. Postel. User Datagram Protocol. IETF RFC 768 (Standard), Αύγουστος 1980.

[Pos81a] J. Postel. Internet Protocol. IETF RFC 791 (Standard), Σεπτέμβριος 1981. Αντικαταστάθηκε από το RFC 1348.

[Pos81b] J. Postel. Transmission Control Protocol. IETF RFC 793 (Standard), Σεπτέμβριος 1981. Αντικαταστάθηκε από τα RFCs 1122, 3168, 6093.

[RLH06] Y. Rekhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). RFC 4271, Ιανουάριος 2006. Αντικαταστάθηκε από το RFC 6286.

[RHW11] C. Raiciu, M. Handley, and D. Wischik. Coupled Congestion Control for Multipath Transport Protocols. IETF RFC 6356, Οκτώβριος 2011.

[ROA05] K. Rojviboonchai, T. Osuga, and H. Aida. R-M/TCP: Protocol for Reliable Multi-Path Transport over the Internet. In *International Conference on Advanced Information Networking and Applications (AINA)*, 19th, σελίδες 801–806. IEEE, Μάρτιος 2005.

[SF12] M. Scharf and A. Ford. MPTCP Application Interface Considerations, Internet draft, draft-ietf-mptcp-api-05.txt, Απρίλιος 2012.

[Ste97] W. Stevens. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. RFC 2001 (Proposed Standard), Ιανουάριος 1997. Αντικαταστάθηκε από το RFC 2581.

[WRGH11] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley. Design, implementation and evaluation of congestion control for multipath TCP. In *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 8th, Απρίλιος 2011.

[ZLK04] M. Zhang, Junwen Lai, and A. Krishnamurthy. A transport layer approach for improving end-to-end performance and robustness using redundant paths. In *USENIX*, σελίδες 99–112, Ιούνιος 2004.