

ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ II

REPORT

ΜΟΥΡΑΤΙΔΗΣ ΑΝΑΣΤΑΣΙΟΣ 9040
ΝΟΕΜΒΡΙΟΣ 2020 | atmourat@ece.auth.gr

Statement of Originality

Δηλώνω υπεύθυνα, ότι εγώ, ο Αναστάσιος Μουρατίδης με ΑΕΜ 9040 και email atmourat@ece.auth.gr, συνέταξα την συγκεκριμένη εργασία ο ίδιος. Ξεκίνησα με την υλοποίηση της echo συνάρτησης, η οποία δέχεται echo packets από την Ιθάκη, στηριζόμενος στον seed code που υπάρχει στο τέλος της εκφώνησης της εργασίας.

Παρατηρήσεις για τα διαγράμματα

Πρώτο session

Εργασία σε δικτυακό προγραμματισμό : Java socket programming	
Client public address	: 46.103.141.95
Client listening port	: 48011
Server public address	: 155.207.18.208
Server listening port	: 38011
Echo request code	: E5163
Image request code	: M9113
Audio request code	: A8803 Ad-hoc real-time audio streaming
Ithakicopter code	: Q9884 Remote flying micro-platform
Vehicle OBD-II code	: V5641 Onboard car fault diagnostics
Διάρκεια εργαστηρίου από 12:17 έως 14:17 σήμερα 23-11-2020	
Αριθμός φοιτητών/φοιτητριών online στο εργαστήριο : 7	

G1, G2:

Σε διάστημα 5 λεπτών στάλθηκαν 188 πακέτα, και παρατηρούμε ότι η καθυστέρηση που εισάγει η Ιθάκη είναι σχεδόν πάντα μεγαλύτερη των 750 ms. Το throughput υπολογίστηκε με την τεχνική του κινούμενου μέσου όρου με παράθυρο 8 sec και είναι μεταξύ 110 – 220 packets/8 sec. Το κάθε πακέτο έχει μήκος 32 bytes, και ένα 1 byte ισούται με 8 bits, οπότε η τιμή του throughput αντιστοιχεί σε 3520 – 7040bps.

G3, G4:

Σε διάστημα 5 λεπτών στάλθηκαν 963 πακέτα. Στο διάγραμμα G3 παρατηρούμε την πραγματική καθυστέρηση μεταξύ server και του client, η οποία είναι περίπου στα 300 *ms*, αλλά και κάποια peaks στα 900 *ms* τα οποία ίσως οφείλονται στο γεγονός ότι λαμβάνω τα πακέτα στον υπολογιστή μου μέσω σύνδεσης WiFi. . Το throughput υπολογίστηκε με την τεχνική του κινούμενου μέσου όρου με παράθυρο 8 *sec* και είναι συνήθως στα 832 *packets/8 sec*. Το κάθε πακέτο έχει μήκος 32 *bytes*, και ένα 1 *byte* ισούται με 8 *bits*, οπότε η τιμή του throughput αντιστοιχεί σε 26624 *bps*.

G5:

Από το ιστόγραμμα του χρόνου απόκρισης, παρατηρούμε ότι η καθυστέρηση που εισάγει η Ιθάκη ακολουθεί κανονική κατανομή με μέση τιμή στα 1599.8 *ms* και διασπορά 19955.

G6:

Από το ιστόγραμμα της ρυθμαπόδοσης, το throughput φαίνεται να προσεγγίζει την κανονική κατανομή.

G7:

Από το ιστόγραμμα του χρόνου απόκρισης, η καθυστέρηση μεταξύ server και client φαίνεται να ακολουθεί εκθετική κατανομή.

G8:

Από αυτό το διάγραμμα φαίνεται ότι η ρυθμαπόδοση στο σύστημα χωρίς καθυστέρηση, παίρνει πιο συχνά την τιμή 832 *packets/8 sec*.

R1:

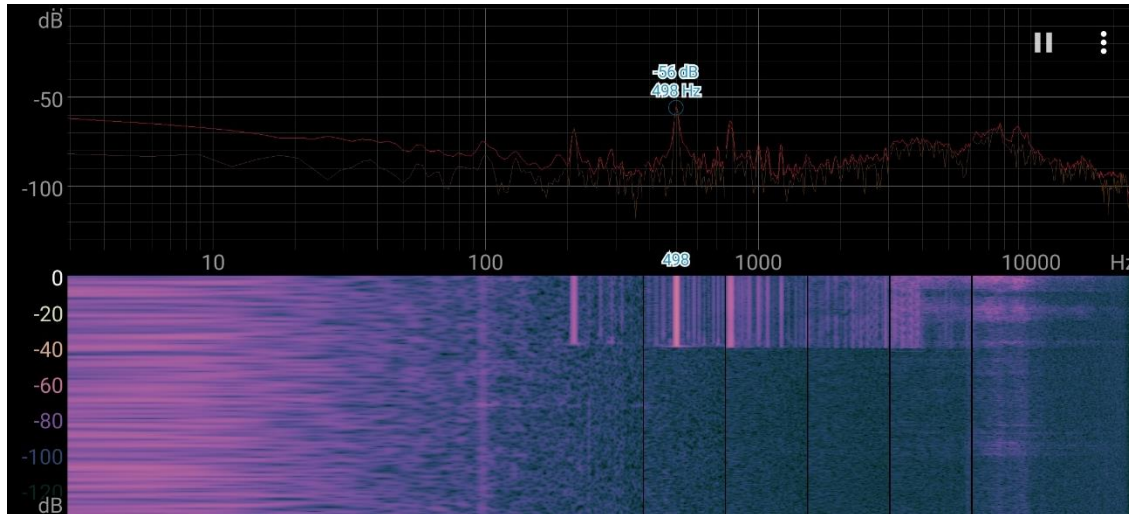
Από αυτό το διάγραμμα παρατηρούμε ότι το SRTT κυμαίνεται γύρω από το 1500, ενώ το RTO παίρνει τιμές μεταξύ 1500 και 4000 με μέση τιμή 3143.

E1, E2:

Η πρώτη φωτογραφία είναι από την κάμερα FIX, ενώ η δεύτερη είναι από την κάμερα PTZ.

G9:

Στο διάγραμμα G9 φαίνεται ένα κομμάτι της κυματομορφής που προκύπτει από γεννήτρια συχνοτήτων. Χρησιμοποιήθηκε η εφαρμογή Spectroid στο smartphone για να αναγνωρίσω την συχνότητα την οποία “γεννά” η Ιθάκη. Η συχνότητα που εντοπίζει η εφαρμογή φαίνεται στην επόμενη φωτογραφία να είναι στα 498 Hz.



G10a, G10b:

Στο G10a απεικονίζεται ένα κομμάτι της κυματομορφής “Stereo Love” και έχει διαμορφωθεί με βάση το DPCM. Στο διάγραμμα G10b φαίνεται κομμάτι της κυματομορφής του ίδιου τραγουδιού αλλά με διαμόρφωση AQDPCM.

G11:

Από το διάγραμμα G11 οι τιμές των διαφορών στην διαμόρφωση DPCM φαίνεται να ακολουθεί τριγωνική κατανομή.

G12:

Από το διάγραμμα G12 οι τιμές των διαφορών στην διαμόρφωση AQDPCM φαίνεται να ακολουθεί Laplace κατανομή.

G13, G14:

Στο διάγραμμα G13 παρατηρούμε ότι οι τιμές των δειγμάτων στην διαμόρφωση DPCM κυμαίνονται από το -20 έως το 15, ενώ οι τιμές των δειγμάτων στην διαμόρφωση AQDPCM κυμαίνονται από -120 έως 130.

G15, G16, G17, G18:

Στα διαγράμματα G15, G16 παρουσιάζονται οι τιμές της μέσης τιμής και του βήματος του προσαρμοσμένου κβαντιστή AQDPCM για το τραγούδι "Stereo Love". Αντίστοιχα στα διαγράμματα G17, G18 παρουσιάζονται οι τιμές της μέσης τιμής και του βήματος του προσαρμοσμένου κβαντιστή AQDPCM για το τραγούδι "Sexy Love". Δεν φαίνεται σε κάποια από τα παραπάνω διαγράμματα να ακολουθείται κάποια γνωστή κατανομή.

G19:

Από το διάγραμμα G19 για αίτημα FLIGHTLEVEL = 100, LMOTOR = 120, RMOTOR = 120, τα PRESSURE, ALTITUDE και TEMPERATURE παραμένουν σταθερά ενώ οι δύο MOTOR πιάνουν την τιμή 120 και στη συνέχεια πέφτει σταδιακά η τιμή τους στη τιμή 0.

G20:

Από το διάγραμμα G20 για αίτημα FLIGHTLEVEL = 200, LMOTOR = 200, RMOTOR = 200, τα PRESSURE και TEMPERATURE παραμένουν σταθερά ενώ το ALTITUDE αυξάνεται και στη συνέχεια μειώνεται. Τα RMOTOR και LMOTOR έχουν σχεδόν σταθερή τιμή σχεδόν καθ' όλη τη διάρκεια επικοινωνίας της εφαρμογής με την Ιθάκη με μια μικρή σταδιακή μείωση.

OBD1:

Η θερμοκρασία του αέρα εισόδου αυξάνεται από 12 σε 14 κατά τη λειτουργία της μηχανής στο διάστημα των 4 λεπτών.

OBD2:

Η θέση του γκαζιού κυμαίνεται γύρω από το 16% με κάποιες μικρές αυξομειώσεις αλλά και μία απότομη στιγμιαία τιμή στο 100%.

OBD3:

Οι στροφές της μηχανής παίρνουν τιμές ανάμεσα στις 1000-2000 rpm.

OBD4:

Η ταχύτητα του οχήματος παίρνει τιμές από 0-55 km/h, εκτός από μια ακραία τιμή στα 255 km/h.

OBD5:

Η θερμοκρασία του ψυκτικού υγρού αυξάνεται από τους 10 °C στους 62°C.

Δεύτερο session

Εργασία σε δικτυακό προγραμματισμό : Java socket programming	
Client public address	: 46.103.141.95
Client listening port	: 48039
Server public address	: 155.207.18.208
Server listening port	: 38039
Echo request code	: E0661
Image request code	: M7273
Audio request code	: A9035 Ad-hoc real-time audio streaming
Ithakicopter code	: Q0287 Remote flying micro-platform
Vehicle OBD-II code	: V1680 Onboard car fault diagnostics
Διάρκεια εργαστηρίου από 15:19 έως 17:19 σήμερα 25-11-2020	
Αριθμός φοιτητών/φοιτητριών online στο εργαστήριο : 17	

G1, G2:

Σε διάστημα 5 λεπτών στάλθηκαν 189 πακέτα, και παρατηρούμε ότι η καθυστέρηση που εισάγει η Ιθάκη είναι σχεδόν πάντα μεγαλύτερη των 600 ms. Το throughput υπολογίστηκε με την τεχνική του κινούμενου μέσου όρου με παράθυρο 8 sec και είναι περίπου μεταξύ 96 – 256 packets/8 sec. Το κάθε πακέτο έχει μήκος 32 bytes, και ένα 1 byte ισούται με 8 bits, οπότε η τιμή του throughput αντιστοιχεί σε 3072 – 8192 bps.

G3, G4:

Σε διάστημα 5 λεπτών στάλθηκαν 965 πακέτα. Στο διάγραμμα G3 παρατηρούμε την πραγματική καθυστέρηση μεταξύ server και του client, η οποία είναι περίπου στα 300 ms. Το throughput υπολογίστηκε με την τεχνική του κινούμενου μέσου όρου με παράθυρο 8 sec και είναι συνήθως στα 832 packets/8 sec. Το κάθε πακέτο έχει μήκος 32 bytes, και ένα 1 byte ισούται με 8 bits, οπότε η τιμή του throughput αντιστοιχεί σε 26624 bps.

G5:

Από το ιστόγραμμα του χρόνου απόκρισης, παρατηρούμε ότι η καθυστέρηση που εισάγει η Ιθάκη ακολουθεί κανονική κατανομή με μέση τιμή στα 1588.2 ms και διασπορά 30420.

G6:

Από το ιστόγραμμα της ρυθμαπόδοσης, το throughput φαίνεται να προσεγγίζει την τριγωνική κατανομή.

G7:

Από το ιστόγραμμα του χρόνου απόκρισης, η καθυστέρηση μεταξύ server και client συγκεντρώνεται κυρίως στην τιμή 300 και από το συγκεκριμένο διάγραμμα δεν φαίνεται να ακολουθεί κάποια γνωστή κατανομή.

G8:

Από αυτό το διάγραμμα φαίνεται ότι η ρυθμαπόδοση στο σύστημα χωρίς καθυστέρηση, παίρνει πιο συχνά την τιμή $832\text{ packets}/8\text{ sec}$.

R1:

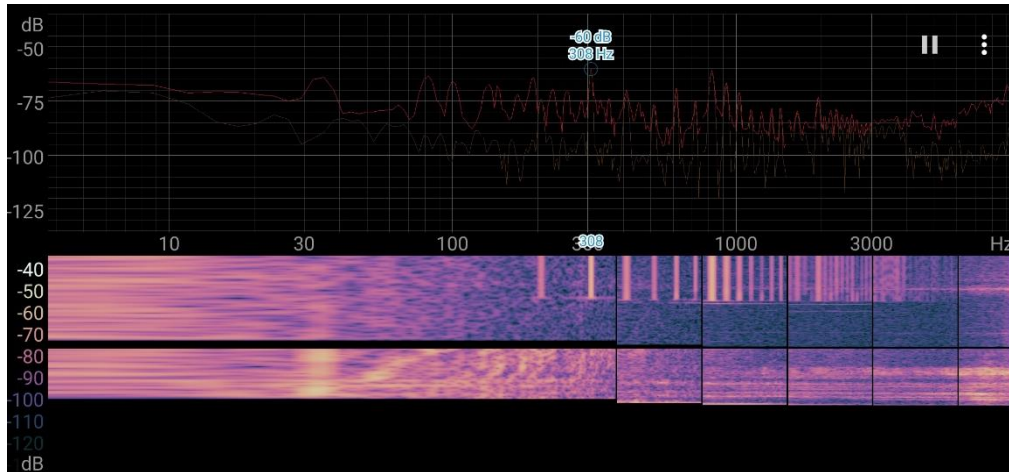
Από αυτό το διάγραμμα παρατηρούμε ότι το SRTT κυμαίνεται γύρω από το 1800, ενώ το RTO παίρνει τιμές μεταξύ 2000 και 4900 με μέση τιμή 3403.

E1, E2:

Η πρώτη φωτογραφία είναι από την κάμερα FIX, ενώ η δεύτερη είναι από την κάμερα PTZ.

G9:

Στο διάγραμμα G9 φαίνεται ένα κομμάτι της κυματομορφής που προκύπτει από γεννήτρια συχνοτήτων. Χρησιμοποιήθηκε η εφαρμογή Spectroid στο smartphone για να αναγνωρίσω την συχνότητα την οποία “γεννά” η Ιθάκη. Η συχνότητα που εντοπίζει η εφαρμογή φαίνεται στην επόμενη φωτογραφία να είναι στα 308 Hz . Ένα ακόμα peak παρατηρείται και κοντά στα 800 Hz , οπότε από την γεννήτρια προκύπτει μια σύνθεση συχνοτήτων.



G10a, G10b:

Στο G10a απεικονίζεται ένα κομμάτι της κυματομορφής “Losing My Religion” και έχει διαμορφωθεί με βάση το DPCM. Στο διάγραμμα G10b φαίνεται κομμάτι της κυματομορφής του ίδιου τραγουδιού αλλά με διαμόρφωση AQDPCM.

G11:

Από το διάγραμμα G11 οι τιμές των διαφορών στην διαμόρφωση DPCM φαίνεται να ακολουθεί τριγωνική κατανομή.

G12:

Από το διάγραμμα G12 οι τιμές των διαφορών στην διαμόρφωση AQDPCM φαίνεται να ακολουθεί Laplace κατανομή.

G13, G14:

Στο διάγραμμα G13 παρατηρούμε ότι οι τιμές των δειγμάτων στην διαμόρφωση DPCM κυμαίνονται από το -7 έως το 8, ενώ οι τιμές των δειγμάτων στην διαμόρφωση AQDPCM κυμαίνονται από -130 έως 110.

G15, G16, G17, G18:

Στα διαγράμματα G15, G16 παρουσιάζονται οι τιμές της μέσης τιμής και του βήματος του προσαρμοσμένου κβαντιστή AQDPCM για το τραγούδι “Losing My Religion”. Αντίστοιχα στα διαγράμματα G17, G18 παρουσιάζονται οι τιμές της μέσης τιμής και του βήματος του προσαρμοσμένου κβαντιστή AQDPCM για το τραγούδι “Το τραγούδι της ξενιτιάς”. Δεν φαίνεται σε κάποια από τα παραπάνω διαγράμματα να ακολουθείται κάποια γνωστή κατανομή.

G19:

Από το διάγραμμα G19 για αίτημα FLIGHTLEVEL = 200, LMOTOR = 200, RMOTOR = 200, τα PRESSURE και TEMPERATURE παραμένουν σταθερά ενώ το ALTITUDE αυξάνεται και στη συνέχεια μειώνεται. Τα RMOTOR και LMOTOR έχουν σχεδόν σταθερή τιμή σχεδόν καθ' όλη τη διάρκεια επικοινωνίας της εφαρμογής με την Ιθάκη με μια μικρή σταδιακή μείωση.

G20:

Από το διάγραμμα G20 για αίτημα FLIGHTLEVEL = 150, LMOTOR = 150, RMOTOR = 150, τα PRESSURE, ALTITUDE και TEMPERATURE παραμένουν σταθερά ενώ οι δύο MOTOR πιάνουν την τιμή 150 και στη συνέχεια πέφτει σταδιακά η τιμή τους στη τιμή 0.

OBD1:

Η θερμοκρασία του αέρα εισόδου αυξάνεται από 12 σε 14 κατά τη λειτουργία της μηχανής στο διάστημα των 4 λεπτών.

OBD2:

Η θέση του γκαζιού κυμαίνεται γύρω από το 16% με κάποιες μικρές αυξομειώσεις.

OBD3:

Οι στροφές της μηχανής παίρνουν τιμές ανάμεσα στις 1000-2000 rpm.

OBD4:

Η ταχύτητα του οχήματος παίρνει τιμές από 0-55 km/h, εκτός από μια ακραία τιμή στα 255 km/h.

OBD5:

Η θερμοκρασία του ψυκτικού υγρού αυξάνεται από τους 10 °C – 62°C.

Σχόλια για το Wireshark

Τα σχόλια που ακολουθούν αφορούν το **δεύτερο session** ενδεικτικά, αλλά προφανώς ισχύουν και για το πρώτο. Όσον αφορά την ενότητα **Echo**, στην πρώτη εικόνα έχουμε «πατημένο» ένα πακέτο μήκους 5 και πρόκειται για το request code που στέλνει η εφαρμογή στην Ιθάκη, ενώ στη δεύτερη ένα πακέτο μήκους 32 που είναι το echo πακέτο που στέλνει η Ιθάκη. Στην ενότητα **Image** στην πρώτη και στην τρίτη εικόνα βλέπουμε τα δύο request που στέλνουμε στην Ιθάκη μήκους 13 και 16 για κάμερες FIX (αν δεν γράψουμε παίρνουμε by default από την κάμερα FIX) και PTZ αντίστοιχα. Στις εικόνες 2 και 4 βλέπουμε τόσο πακέτα εικόνας μήκους 1024 bytes όσο και τα τελευταία πακέτα μήκους 331 και 409 αντίστοιχα (το τελευταίο πακέτο της εικόνας έχει διαφορετικό μήκος). Στην ενότητα **Audio**, παρατηρούμε τα πακέτα μήκους 128 που προέχονται από διαμόρφωση DPCM καθώς και τα πακέτα μήκους 132 που προέχονται από διαμόρφωση AQDPCM. Στην ενότητα **Ithakicopter** σημειώνουμε ότι η επικοινωνία γίνεται με το πρωτόκολλο TCP όπως φαίνεται και στις εικόνες. Τέλος ακολουθούν screenshots από την επικοινωνία της εφαρμογής μας με το **OBD II**.

Πρωτόκολλο UDP

Το πρωτόκολλο User Datagram Protocol (UDP) είναι ένα από τα βασικά πρωτόκολλα που χρησιμοποιούνται στο διαδίκτυο. Ένα από τα κύρια χαρακτηριστικά του UDP είναι ότι δεν εγγυάται αξιόπιστη επικοινωνία. Τα πακέτα UDP που αποστέλλονται από έναν υπολογιστή μπορεί να φτάσουν στον παραλήπτη με λάθος σειρά, διπλά ή να μην φτάσουν καθόλου εάν το δίκτυο έχει μεγάλο φόρτο. Χρησιμοποιείται όταν η "γρήγορη" παράδοση των πακέτων είναι πιο σημαντική από την "ακριβή" παράδοση, πχ στη μετάδοση ομιλίας και βίντεο. Το UDP πρόκειται για ένα ασυνδεδεσμένο πρωτόκολλο καθώς δεν δημιουργεί συνδέσεις μεταξύ των εφαρμογών αλλά στέλνει μόνο πακέτα. Επιπλέον, το UDP υποστηρίζει broadcasting.

Δομή UDP πρωτοκόλλου

Το UDP βρίσκεται ανάμεσα στο επίπεδο δικτύου (network layer) και στο επίπεδο συνόδου (session layer) ή εφαρμογών (application layer).

Κάθε πακέτο UDP έχει μία κεφαλίδα (header) που αναφέρει τα χαρακτηριστικά του. Η κεφαλίδα περιλαμβάνει μονάχα 4 πεδία, τα οποία είναι πολύ λίγα εάν συγκριθούν με άλλα πρωτόκολλα, όπως το TCP. Δύο από τα τέσσερα πεδία είναι προαιρετικά (φαίνονται χρωματισμένα με ροζ).

+	Bits 0 - 15	16 - 31
0	Source Port	Destination Port
32	Length	Checksum
64	Data	

Ακολουθεί μία συνοπτική εξήγηση των πεδίων:

- **Source port**

Η πόρτα του αποστολέα από την οποία προήλθε το πακέτο. Εάν ο παραλήπτης επιθυμεί να στείλει κάποια απάντηση, θα πρέπει να την στείλει στην πόρτα αυτήν. Το συγκεκριμένο πεδίο δεν είναι υποχρεωτικό και στις περιπτώσεις που δεν χρησιμοποιείται θα πρέπει να έχει την τιμή μηδέν.

- **Destination port**

Η πόρτα του παραλήπτη στην οποία θα πρέπει να παραδοθεί το πακέτο.

- **Length**

Το πεδίο αυτό έχει μέγεθος 16-bit και περιλαμβάνει το μέγεθος του πακέτου σε bytes. Το μικρότερο δυνατό μέγεθος είναι 8 bytes, αφού η κεφαλίδα αυτή καθ' αυτή καταλαμβάνει τόσο χώρο. Θεωρητικά, το μέγεθος του UDP πακέτου δεν μπορεί να ξεπερνάει τα 65,527 bytes, αλλά πρακτικά το όριο μειώνεται στα 65,507 bytes λόγω διαφόρων περιορισμών που εισάγει το πρωτόκολλο IPv4 στο επίπεδο δικτύου.

- **Checksum**

Ένα πεδίο 16-bit το οποίο χρησιμοποιείται για επαλήθευση της ορθότητας του πακέτου στο σύνολό του, δηλαδή τόσο της κεφαλίδας όσο και των δεδομένων.

Στην συνέχεια το πακέτο UDP περνάει στο επίπεδο δικτύου, το οποίο αναλαμβάνει να το μεταδώσει στο δίκτυο υπολογιστών. Το επίπεδο αυτό τοποθετεί μία ακόμη κεφαλίδα στο πακέτο, η οποία διαφέρει ανάλογα με την έκδοση του πρωτοκόλλου που χρησιμοποιείται στο επίπεδο δικτύου (IPv4 ή IPv6).

IPv4

Για IPv4, το πακέτο λαμβάνει την ακόλουθη μορφή:

+	Bits 0 - 7	8 - 15	16 - 23	24 - 31
0	Source address			
32	Destination address			
64	Zeros	Protocol	Length	
96	Source Port		Destination Port	
128	Length		Checksum	
160	Data			

- **Source Address, Destination Address**

Οι διευθύνσεις IP του αποστολέα και του παραλήπτη αντίστοιχα.

- **Zeros**

Μία ακολουθία μηδενικών, η οποία δεν παίζει κανέναν ρόλο κατά την μετάδοση του πακέτου.

- **Protocol**

Ένας χαρακτηριστικός αριθμός που αντιστοιχεί στο πρωτόκολλο που χρησιμοποιείται. Για το UDP η τιμή που παίρνει το πεδίο αυτό είναι 17.

- **Length**

Το συνολικό μέγεθος του πακέτου. Στην συγκεκριμένη περίπτωση UDP length + IP Header length.

IPv6

Για IPv6, το πακέτο παίρνει την εξής μορφή:

+	Bits 0 - 7	8 - 15	16 - 23	24 - 31
0	Source address			
32				
64				
96				
128	Destination address			
160				
192				
256				
288	UDP length			
320	Zeros			Next Header
352	Source Port		Destination Port	
384	Length		Checksum	
416	Data			

- **Source Address, Destination Address**

Οι διευθύνσεις IP του αποστολέα και του παραλήπτη αντίστοιχα, οι οποίες όμως στην περίπτωση αυτή είναι τύπου IPv6, δηλαδή πολύ μεγαλύτερες (IPv4 - 32bit, IPv6 - 128bit).

- **UDP Length**

Το συνολικό μέγεθος του πακέτου UDP, όπως και προηγουμένως.

- **Zeros**

Μία ακολουθία μηδενικών, η οποία δεν παίζει κανέναν ρόλο κατά την μετάδοση του πακέτου.

- **Next Header**

Το πεδίο αυτό παίρνει μία τιμή που είναι χαρακτηριστική για το πρωτόκολλο που χρησιμοποιείται. Στην περίπτωση του UDP, η τιμή αυτή είναι 17.

Στην περίπτωση IPv6 το πεδίο checksum του UDP πακέτου δεν είναι πλέον προαιρετικό, αλλά θα πρέπει υποχρεωτικά να συμπληρωθεί.

Εφαρμογές UDP

Το UDP χρησιμοποιείται όταν η γρήγορη παράδοση των πακέτων είναι πιο σημαντική από την ακριβή παράδοση. Χαρακτηριστικό παράδειγμα αποτελούν οι εφαρμογές audio και video streaming. Για τις εφαρμογές αυτές προτιμάται το πρωτόκολλο UDP διότι είναι αρκετά γρήγορο, και πιο αποτελεσματικό συγκριτικά με το TCP μιας και δεν απαιτούν αξιόπιστη επικοινωνία. Αυτό συμβαίνει γιατί, στην περίπτωση που χαθεί κάποιο πακέτο, οι εφαρμογές αυτές διαθέτουν ειδικούς μηχανισμούς διόρθωσης και παρεμβολής ούτως ώστε ο τελικός χρήστης να μην παρατηρεί καμία αλλοίωση ή διακοπή στην ροή του ήχου και της εικόνας λόγω του χαμένου πακέτου.

Μερικές σημαντικές εφαρμογές που χρησιμοποιούν πακέτα UDP είναι οι εξής: Domain Name System (DNS), IPTV, Voice over IP (VoIP), Trivial File Transfer Protocol (TFTP) και τα παιχνίδια που παίζονται ζωντανά μέσω του Διαδικτύου.

Σύγκριση TCP και UDP

Το πρωτόκολλο TCP λειτουργεί εγκαθιδρύοντας συνδέσεις μεταξύ του αποστολέα και του παραλήπτη των πακέτων. Από την στιγμή που μία σύνδεση εγκαθιδρυθεί με επιτυχία, όλα τα δεδομένα αποστέλλονται από τον έναν υπολογιστή στον άλλο με την μορφή πακέτων χρησιμοποιώντας την σύνδεση αυτή. Τα κύρια χαρακτηριστικά του TCP είναι τα εξής:

- **Αξιοπιστία** - Το TCP χρησιμοποιεί διάφορους μηχανισμούς ούτως ώστε να διασφαλιστεί ότι τα πακέτα που μεταδίδονται από τον αποστολέα θα φτάσουν σίγουρα στον παραλήπτη και στην σωστή σειρά. Οι μηχανισμοί αυτοί περιλαμβάνουν την επιβεβαίωση λήψης πακέτου από τον παραλήπτη, την επαναποστολή πακέτων που χάθηκαν και τον καθορισμό ενός ελάχιστου

χρονικού διαστήματος μέσα στο οποίο κάθε αποτελούμενο πακέτο θα πρέπει να έχει παραληφθεί (timeout). Στην περίπτωση που χαθεί κάποιο πακέτο, ο αποστολέας προσπαθεί και πάλι να το ξαναστείλει. Επίσης, εάν ο παραλήπτης διαπιστώσει ότι ένα πακέτο δεν του έχει έρθει, τότε θα ζητήσει από τον αποστολέα να του το ξαναστείλει.

- **Σειρά πακέτων** - Εάν δύο πακέτα αποσταλούν σε μία σύνδεση το ένα μετά το άλλο, τότε το πρωτόκολλο TCP εγγυάται ότι θα φτάσουν στον παραλήπτη με την ίδια σειρά με την οποία στάλθηκαν. Στην περίπτωση που λείπει ένα πακέτο και έρθουν μελλοντικά πακέτα, τότε αυτά κατακρατούνται στην προσωρινή μνήμη (buffer) μέχρις ότου φτάσει το πακέτο που λείπει. Τότε αναδιατάσσονται και εμφανίζονται με την σωστή σειρά στον παραλήπτη.
- **Βαρύτητα** - Το πρωτόκολλο TCP θεωρείται ιδιαίτερα βαρύ, δεδομένου του γεγονότος ότι χρειάζονται τουλάχιστον 3 πακέτα για την εγκαθίδρυση της σύνδεσης, πριν ακόμη μεταδοθεί οποιοδήποτε πακέτο δεδομένων. Επίσης, οι μηχανισμοί αξιοπιστίας που υλοποιεί το κάνουν ακόμη πιο βαρύ, πράγμα που έχει φυσικά σημαντικό αντίκτυπο στην ταχύτητα μετάδοσης δεδομένων.

Το UDP είναι ένα πιο απλό και ελαφρύ πρωτόκολλο, στο οποίο δεν υπάρχει η έννοια της σύνδεσης. Κάθε πακέτο UDP διανύει το δίκτυο ως μία ξεχωριστή αυτόνομη μονάδα και όχι ως μία σειρά πακέτων σε μία σύνδεση, όπως στο TCP. Τα κύρια χαρακτηριστικά του UDP είναι τα εξής:

- **Αναξιόπιστο** - Κατά την αποστολή ενός πακέτου, ο αποστολέας δεν είναι σε θέση να γνωρίζει εάν το πακέτο θα φτάσει σωστά στον προορισμό του ή εάν θα χαθεί μέσα στο δίκτυο. Δεν έχει προβλεφθεί η δυνατότητα επιβεβαίωσης λήψης πακέτου από τον παραλήπτη, ούτε η επαναμετάδοση ενός χαμένου πακέτου.
- **Δεν υπάρχει σειρά** - Τα πακέτα UDP, σε αντίθεση με το TCP, δεν αριθμούνται και κατά συνέπεια δεν υπάρχει κάποια συγκεκριμένη σειρά με την οποία θα πρέπει να φτάσουν στον παραλήπτη.
- **Ελαφρύ** - Το πρωτόκολλο αυτό καθ' αυτό είναι πολύ ελαφρύ σε σύγκριση με το TCP διότι δεν εφαρμόζει όλους τους μηχανισμούς αξιοπιστίας επικοινωνίας που υπάρχουν στο δεύτερο. Αυτό έχει ως συνέπεια να είναι αρκετά πιο γρήγορο.
- **Datagrams** - Κάθε πακέτο UDP ονομάζεται επίσης και "datagram", θεωρείται δε ως μεμονωμένη οντότητα που θα πρέπει να μεταδοθεί ολόκληρη. Κατά συνέπεια δεν υφίσταται η έννοια της διοχέτευσης πακέτων μέσα σε ένα κανάλι/σύνδεση.

Διεθνή Πρότυπα Audio Streaming

Εισαγωγικά αναφέρουμε ότι η ροή ήχου συμπιέζεται για να μειωθεί το μέγεθος του αρχείου χρησιμοποιώντας μια μορφή κωδικοποίησης ήχου όπως MP3, Vorbis, AAC ή Opus. Το bitstream παρέχεται από ένα διακομιστή σε έναν πελάτη (π.χ. ο χρήστης με το φορητό υπολογιστή συνδεδεμένο στο διαδίκτυο) χρησιμοποιώντας ένα πρωτόκολλο μεταφοράς, όπως το RTMP ή το RTP της Adobe. Στη δεκαετία του 2010 εμφανίστηκαν τεχνολογίες όπως το HLS της Apple, το Smooth Streaming της Microsoft, το HDS της Adobe και οι μη αποκλειστικές (ιδιόκτητες) μορφές, όπως το MPEG-DASH, για να επιτρέψουν την προσαρμοστική ροή bitrate μέσω HTTP ως εναλλακτική λύση στη χρήση ιδιόκτητων πρωτοκόλλων μεταφοράς. Συχνά, ένα πρωτόκολλο μεταφοράς συνεχούς ροής χρησιμοποιείται για την αποστολή ήχου από ένα χώρο εκδήλωσης σε μια υπηρεσία κωδικοποίησης "cloud" και στο CDN, το οποίο στη συνέχεια χρησιμοποιεί πρωτόκολλα μεταφοράς HTTP για τη διανομή του βίντεο σε μεμονωμένα σπίτια και χρήστες. Ο πελάτης ροής (ο τελικός χρήστης) μπορεί να αλληλεπιδράσει με τον διακομιστή συνεχούς ροής χρησιμοποιώντας ένα πρωτόκολλο ελέγχου, όπως το MMS ή το RTSP.

Ακολουθούν κάποια από τα πιο διαδεδομένα πρότυπα που χρησιμοποιούνται για audio streaming.

RTP – Real Time Protocol

Το RTP είναι ένα πρωτόκολλο βασισμένο στο IP που παρέχει υποστήριξη για τη μεταφορά των δεδομένων σε πραγματικό χρόνο, όπως ροές ήχου και βίντεο. Οι υπηρεσίες που παρέχονται από το RTP περιλαμβάνουν την ανασυγκρότηση του χρόνου, την ανίχνευση των απωλειών, την ασφάλεια και την ταυτοποίηση του περιεχομένου. Το RTP έχει σχεδιαστεί κυρίως για πολυεκπομπή (multicast) των δεδομένων σε πραγματικό χρόνο, αλλά μπορεί να χρησιμοποιηθεί επίσης σε unicast. Μπορεί να χρησιμοποιηθεί για τη μονόδρομη μεταφορά, video-on-demand, καθώς και για διαδραστικές υπηρεσίες, όπως η τηλεφωνία μέσω του διαδικτύου. Επιπλέον, παρέχει μια κοινή πλατφόρμα για τη μεταφορά δεδομένων και την έκφραση πληροφοριών συγχρονισμού που απαιτούνται από εφαρμογές πραγματικού χρόνου.

Το RTP έχει σχεδιαστεί για να λειτουργεί σε συνδυασμό με το βοηθητικό RTCP πρωτόκολλο ελέγχου για να πάρει πληροφορίες σχετικά με την ποιότητα μετάδοσης

δεδομένων και των πληροφοριών σχετικά με τους συμμετέχοντες στη σύνοδο που βρίσκεται σε εξέλιξη.

Το RTP δεν παρέχει όλη τη λειτουργικότητα που παρέχεται από ένα τυπικό πρωτόκολλο μεταφοράς. Επίσης, δεν είναι connection-oriented μιας και μπορεί να λειτουργεί πάνω από connectionoriented πρωτόκολλα ή πάνω από χαμηλού επιπέδου connection-less πρωτόκολλα). Παρέχει από άκρο σε άκρο (end-to-end) υπηρεσίες παράδοσης δεδομένων με χαρακτηριστικά πραγματικού χρόνου, όπως διαδραστικά βίντεο και ήχο. Ωστόσο το ίδιο το RTP δεν παρέχει κανένα μηχανισμό για να εξασφαλιστεί η έγκαιρη παράδοση. Χρειάζεται υποστήριξη από τα χαμηλότερα στρώματα που έχουν ουσιαστικά τον έλεγχο των πόρων σε switches και routers. Το RTP εξαρτάται από το RSVP για να δεσμεύσει τους πόρους και να παράσχει την απαιτούμενη ποιότητα της υπηρεσίας (QoS).

RTCP - Real Time Control Protocol

Το RTP Control Protocol (RTCP) είναι «αδελφικό» πρωτόκολλο του Realtime Transport Protocol (RTP). Το RTCP παρέχει στατιστικές και πληροφορίες ελέγχου για μια σύνοδο RTP. Συνεργάζεται με το RTP κατά την παράδοση και τη οργάνωση των πακέτων των δεδομένων πολυμέσων, αλλά δεν μεταφέρει κανένα δεδομένο από μόνο του. Η πρωταρχική λειτουργία του RTCP είναι να παρέχει πληροφορίες σχετικά με την ποιότητα της υπηρεσίας (QoS) στην κατανομή των πολυμέσων και να αποστέλλει κατά διαστήματα στατιστικές πληροφορίες για τους συμμετέχοντες σε μια σύνοδο συνεχούς ροής πολυμέσων.

RTSP - Real Time Streaming Protocol

Το RTSP είναι πρωτόκολλο ελέγχου παρουσίασης πολυμέσων στα IP δίκτυα. Αξιοποιεί την υπάρχουσα υποδομή του ιστού και είναι αποτελεσματικό τόσο σε πολυπληθή ακροατήρια όσο και για μεμονωμένους παραλήπτες. Πρόκειται για ένα πρωτόκολλο επιπέδου εφαρμογής με σύνταξη και λειτουργία παρόμοια με του HTTP, αλλά λειτουργεί για ήχο και βίντεο. Χρησιμοποιεί διευθύνσεις URL, όπως αυτές στο HTTP. Υποστηρίζει πολλαπλούς εξυπηρετητές, επομένως παραλήπτης μπορεί να δέχεται μια ροή δεδομένων μέσω πολλαπλών συνόδων με πολλούς εξυπηρετητές ταυτόχρονα. Επίσης, Το RTSP υλοποιείται σε πολλαπλές πλατφόρμες λειτουργικών συστημάτων αφού επιτρέπει τη διαλειτουργικότητα μεταξύ των πελατών και των εξυπηρετητών από διαφορετικούς κατασκευαστές.

Το πρωτόκολλο RTSP εγκαθιστά και ελέγχει είτε ένα είτε αρκετά streams από μέσα όπως ήχος ή βίντεο. Το πρωτόκολλο δεν μεταφέρει τα συνεχόμενα πακέτα από μόνο του, παρόλο που παρεμβολές στα κύματα πακέτων με πακέτα ελέγχου είναι δυνατές. Με άλλα λόγια το RTSP λειτουργεί σαν ένα «τηλεκοντρόλ δικτύου» για servers πολυμέσων.

Το RTSP έχει στόχο να παρέχει τις ίδιες υπηρεσίες σε συνεχή ροή ήχου και βίντεο όπως ακριβώς κάνει το HTTP για κείμενο και γραφικά. Έχει σχεδιαστεί σκόπιμα να έχει παρόμοια σύνταξη και λειτουργία, έτσι ώστε οι περισσότεροι μηχανισμοί επέκτασης του HTTP να μπορούν να προστεθούν στο RTSP. Στο RTSP, κάθε παρουσίαση και ροή πολυμέσων προσδιορίζεται από μια διεύθυνση URL RTSP. Η συνολική παρουσίαση και οι ιδιότητες των μέσων καθορίζονται σε ένα αρχείο περιγραφής παρουσίασης, το οποίο μπορεί να περιλαμβάνει την κωδικοποίηση, τη γλώσσα, τις διευθύνσεις URL RTSP, τη διεύθυνση προορισμού, τη θύρα, και άλλες παραμέτρους. Το αρχείο περιγραφής παρουσίασης μπορεί να ληφθεί από τον πελάτη χρησιμοποιώντας το πρωτόκολλο HTTP, το ηλεκτρονικό ταχυδρομείο ή άλλα μέσα.

Το RTSP ωστόσο διαφέρει από το HTTP από πολλές απόψεις. Πρώτον, ενώ το HTTP είναι ένα πρωτόκολλο χωρίς καταστάσεις, ένας εξυπηρετητής RTSP πρέπει να διατηρεί «καταστάσεις συνεδρίας», προκειμένου να συσχετίζει τις αιτήσεις RTSP με τη ροή πολυμέσων. Δεύτερον, το HTTP είναι βασικά ένα ασύμμετρο πρωτόκολλο όπου ο πελάτης εκδίδει αιτήματα και ο εξυπηρετητής απαντά, αλλά στο RTSP τόσο ο εξυπηρετητής πολυμέσων όσο και ο πελάτης μπορούν να δημιουργήσουν αιτήματα. Για παράδειγμα, ο εξυπηρετητής μπορεί να δημιουργήσει ένα αίτημα για να ρυθμιστεί η αναπαραγωγή των παραμέτρων μιας ροής πολυμέσων.

RSVP

Το RSVP (Resource ReSerVation Protocol) είναι πρωτόκολλο ελέγχου δικτύου που επιτρέπει στον παραλήπτη των δεδομένων να απαιτήσει συγκεκριμένη άκρο-σε-άκρο ποιότητα υπηρεσίας (QoS) για την ροή των δεδομένων. Οι εφαρμογές πραγματικού χρόνου χρησιμοποιούν το RSVP για να εξασφαλίζουν τους απαραίτητους πόρους στους δρομολογητές (routers) ανάμεσα στις διαδρομές μεταφοράς δεδομένων έτσι ώστε το αιτούμενο bandwidth να είναι διαθέσιμο όταν η μεταφορά των δεδομένων λαμβάνει χώρα. Πρέπει να διευκρινιστεί ότι το RSVP πρωτόκολλο δεν είναι υπεύθυνο για την μεταφορά των πολυμεσικών δεδομένων.

Το RSVP πρωτόκολλο διαχωρίζει τους αποστολείς από τους παραλήπτες. Μια RSVP κράτηση μπορεί να διατηρήσει πόρους για την ροή δεδομένων μόνο προς μια κατεύθυνση. Ουσιαστικά μια πολυμεσική μεταφορά δεδομένων χρειάζεται

εξασφάλιση πόρων στο δίκτυο από τον αποστολέα προς τον παραλήπτη των δεδομένων.

Σε μια επικοινωνία μεταξύ ομάδων ετερογενών παραληπτών, οι παραλήπτες έχουν διαφορετικές χωρητικότητες και επίπεδα ποιότητας υπηρεσίας. Στην περίπτωση αυτή ο κάθε παραλήπτης είναι υπεύθυνος για να διαλέξει το δικό του επίπεδο ποιότητας υπηρεσίας, αρχικοποιώντας την κράτηση των πόρων και διατηρώντας την ενεργή όσο διάστημα είναι απαραίτητη. Οι αποστολές διαιρούν τα δεδομένα σε ροές δεδομένων. Κάθε μία από αυτές είναι ξεχωριστή RSVP ροή με διαφορετικό επίπεδο ποιότητα υπηρεσίας. Κάθε RSVP ροή είναι ομογενείς και είναι δυνατό οι αποδέκτες να συμμετάσχουν σε μια ή περισσότερες ροές. Αυτή η προσέγγιση είναι κατάλληλη για τους ετερογενείς παραλήπτες ώστε να αιτηθούν για διαφορετική ποιότητα υπηρεσίας η οποία θα είναι προσαρμοσμένη στην χωρητικότητα τους και τις απαιτήσεις τους.

Έχουν γίνει προσπάθειες για να τρέχει το RSVP και στα δύο πρωτόκολλα IPv4, IPv6. Το RSVP παρέχει αδιαφανή μεταφορά των μηνυμάτων traffic control και policy control για να είναι περισσότερη προσαρμόσιμη στις νέες τεχνολογίες.

Πηγές:

- <https://el.wikipedia.org/wiki/UDP>
- http://conta.uom.gr/conta/ekpaideysh/metaptyxiaka/technologies_diktywn/ergasies/2005/Multimedia%20over%20IP.pdf
- http://telematics.upatras.gr/telematics/system/files/bouras_site/ergasies_foithwn/RealTimeProtocols_margariti.pdf
- Tanenbaum, Andrew S. , "Δίκτυα Υπολογιστών"
- Ιάκωβος Βενιέρης "Δίκτυα Ευρείας Ζώνης: 3η Έκδοση"