

MusicOnCloud: Μια cloud-based υπηρεσία online streaming μουσικής

HY - 335α

Χειμερινό εξάμηνο 2014-2015

Διδάσκουσα: Παπαδοπούλη Μαρία
Υπεύθυνος βοηθός: Σουρλίγκας Μανώλης
surligas@csd.uoc.gr

7 Δεκεμβρίου 2014



1 Εισαγωγή

Το MusicOnCloud είναι μια Cloud-based υπηρεσία, ένας απλουστευμένος mini "spotify", που προσφέρει την δυνατότητα αποθήκευσης αρχείων μουσικής στο cloud και αναπαραγωγής αυτών on-demand. Οι χρήστες μπορούν να στείλουν αιτήματα για την αποστολή νέων αρχείων ή την online streaming ήδη διαθέσιμων αρχείων.

Η υλοποίηση αυτού του project θα σας βοηθήσει να γίνετε οικείοι με δημοφιλή δικτυακές υπηρεσίες όπως είναι η μεταφορά αρχείων,

το streaming, και το cloud-computing. Επίσης θα γνωρίσετε ένα framework αναπαραγωγής πολυμέσων που χρησιμοποιείται ευρέως από τις περισσότερες Linux διανομές.

2 Γενική Περιγραφή

Το σύστημα αποτελείται από έναν server και έναν client. Ο server παρακολουθεί ένα directory στο οποίο βρίσκονται αποθηκευμένα τα τραγούδια. Οι τύποι πολυμέσων που υποστηρίζονται είναι mp3 και flac. Ο server είναι σε θέση να διατηρεί πολλαπλές συνδέσεις με διάφορους clients και να εξυπηρετεί τα αιτήματά τους ταυτόχρονα.

Ο κάθε client συνδέεται με τον server και βάση ενός συνόλου δυνατών εντολών μπορεί να ανατρέξει τα διαθέσιμα τραγούδια που βρίσκονται στον server, να κάνει upload ένα αρχείο μουσικής ή να ζητήσει την αναπαραγωγή ενός αρχείου.

3 Η υλοποίηση του MusicOnCloud Server

Ο MusicOnCloud Server είναι υπεύθυνος για την εξυπηρέτηση διάφορων αιτημάτων από τους clients, καθώς και την λήψη νέων αρχείων μουσικής ή online stream των ήδη διαθέσιμων. Για να το πετύχει αυτό, ο server χρησιμοποιεί έναν directory το οποίο ορίζεται με το -d command line argument. Σε αυτό το directory βρίσκονται ήδη ή αποθηκεύονται τα αρχεία μουσικής.

Αρχικά ο server περιμένει για TCP συνδέσεις στην port που ορίζεται από το -l command line argument. Οι clients συνδέονται με TCP σε αυτήν την port και μπορούν να αποστείλουν μια σειρά από εντολές όπως περιγράφονται από τον παρακάτω πίνακα.

Εντολή	Ορίσματα
LS	-
UPLOAD	file name
PLAY	ID, port
DOWNLOAD	ID
STOP	-

Πιο αναλυτικά:

- **LS:** Η εντολή αυτή επιστρέφει στον client την λίστα με τα τραγούδια που είναι διαθέσιμα στο cloud. Κάθε τραγούδι έχει ένα ID το οποίο είναι ένας *unsigned long int*.
- **UPLOAD:** Η εντολή αυτή ακολουθείται από ένα string σαν όρισμα, με το όνομα του αρχείου μουσικής. Με αυτή την εντολή ο client γνωστοποιεί στον server ότι θέλει να ανεβάσει ένα αρχείο μουσικής. Η μεταφορά του αρχείου γίνεται χρησιμοποιώντας TCP.

- **PLAY:** Η συγκεκριμένη εντολή ακολουθείται από το ID του τραγουδιού, το οποίο θέλει να ακούσει ο client. Ο server ξεκινάει το stream του αρχείου χρησιμοποιώντας UDP. Επίσης με το δεύτερο όρισμα port, ο client ενημερώνει τον server σε ποιο UDP port περιμένει να λάβει τα πακέτα του stream.
- **DOWNLOAD:** Με αυτή την εντολή ο client ζητάει να κατεβάσει ένα αρχείο μουσικής με ένα συγκεκριμένο ID. Η μεταφορά του αρχείου γίνεται με TCP.
- **STOP:** Η εντολή αυτή αποστέλλεται στον client υποδηλώνοντας ότι τελείωσε η αναπαραγωγή του τραγουδιού που άκουγε. Η συγκεκριμένη εντολή θα πρέπει να σταλθεί 5 δευτερόλεπτα μετά την αποστολή του τελευταίου πακέτου του τραγουδιού.

Κάθε μια από τις παραπάνω εντολές, αναπαριστάται ως ένας unsigned int 32-bit αριθμός. Ο κωδικός κάθε εντολής είναι διαθέσιμος μέσω του enumeration *command_t* που βρίσκεται στο αρχείο *music_on_cloud.h*. Προφανώς μπορείτε να επεκτείνετε το υπάρχων σετ εντολών αν το κρίνεται απαραίτητο. Τα ID είναι ένας unsigned int των 64-bit και το port ένας unsigned int των 16-bit.

3.1 Δομή πακέτων εντολών

Κάθε πακέτο εντολής θα πρέπει να έχει την εξής μορφή:

3 bytes	32-bits	64-bits	variable length
CMD	command number	timestamp	parameters

Τα πρώτα 3 bytes περιέχουν τους χαρακτήρες 'C', 'M', 'D', ακολουθούμενοι από τον κωδικό της εντολής. Ένα 8-byte timestamp καθορίζει την χρονική στιγμή αποστολής της εντολής. Ένας βολικός τρόπος που χρησιμοποιείται συχνά, είναι ο αριθμός των seconds από το Epoch time.

Αμέσως μετά ακολουθούν τα ορίσματα της κάθε εντολής. Ανάλογα την εντολή, μπορούν να υπάρχουν διαφορετικού τύπου ορίσματα. Στην περίπτωση που το όρισμα είναι string, θα πρέπει να διαχωρίζεται από τα υπόλοιπα ορίσματα τοποθετώντας το ανάμεσα σε bytes με την τιμή 0x0. Έτσι με αυτό τον τρόπο μπορεί να γίνει ευκολότερα το parsing του.

Σημείωση: Το TCP χρησιμοποιεί το αλγόριθμο Nagle, για να αποφύγει την αποστολή μικρών πακέτων. Αυτό ίσως δημιουργήσει πρόβλημα με την αποστολή μερικών εντολών. Για αυτό τον λόγο προτείνεται να απενεργοποιήσετε τον αλγόριθμο αυτό, χρησιμοποιώντας την συνάρτηση *setsockopt* και το όρισμα *TCP_NODELAY*.

3.2 Αποστολή πακέτων μουσικής

Καθώς οι κάρτες ήχου καταναλώνουν με έναν σταθερό ρυθμό δεδομένα, ο αποστολέας θα πρέπει να περιορίζει τον ρυθμό μετάδοσης

ώστε να μην υπερχειλίζουν οι buffers του receiver. Αυτό μπορεί να επιτευχθεί με την εισαγωγή ενός μικρού delay ανάμεσα στην αποστολή των πακέτων χρησιμοποιώντας την συνάρτηση *nanosleep()*. Το delay αυτό καθορίζεται από το μέγεθος των δεδομένων του πακέτου και το bitrate του αρχείου μουσικής.

Για να βρείτε το bitrate ενός .mp3 ή .flac αρχείου, υπάρχουν υλοποιημένες δύο συναρτήσεις, η *get_mp3_bit_rate()* και *get_flac_bit_rate()*, οι οποίες επιστρέφουν το bitrate του αρχείου σε kbps. Οι συναρτήσεις αυτές βρίσκονται στο αρχείο *media/mediaplayer.h*.

3.3 Server Tasks

1. Μόλις συνδεθεί κάποιος client ο server θα πρέπει να εκτυπώνει το εξής μήνυμα:
Welcome client IP
όπου IP, η IP διεύθυνση του client. Την διεύθυνση αυτή θα πρέπει να την βρείτε **προγραμματιστικά** και όχι από κάποιο μήνυμα που θα αποστέλλει ο client και θα την εμπεριέχει.
2. Ο server θα πρέπει να είναι σε θέση να υποστηρίζει πολλαπλούς clients
3. Για κάθε εντολή που λαμβάνει ο server θα πρέπει να εκτυπώνει το εξής μήνυμα:
COMMAND: desc FROM: IP TIME: time
όπου *desc* η περιγραφή της εντολής, IP η IP διεύθυνση του client και *time* η ώρα αποστολής της εντολής.
4. Κατά το stream ενός αρχείου μουσικής ο server θα πρέπει να ανοίγει ένα καινούργιο UDP socket αποστέλλοντας τα πακέτα στον client και στην port που είχε δηλώσει σαν παράμετρο της εντολής **PLAY**.

4 Η υλοποίηση του MusicOnCloud Client

Ο κάθε client συνδέεται στον cloud server χρησιμοποιώντας την IP και την port στην οποία περιμένει ο server για συνδέσεις. Η IP και η port ορίζονται με τα *-s* και *-p* command line arguments αντίστοιχα. Εκτός από τις εντολές που αναφέρθηκαν προηγουμένως, κάθε client θα πρέπει να είναι σε θέση να κάνει τα εξής:

- Μετά από κάθε upload ή download αρχείου να εκτυπώνεται ο χρόνος της μεταφοράς καθώς και το bandwidth που επιτεύχθηκε.
- Μετά το τέλος της αναπαραγωγής ενός τραγουδιού να εκτυπώνεται το bandwidth του stream, το μέσο jitter των UDP πακέτων καθώς και η τυπική απόκλισή τους.
- Τέλος ο client θα πρέπει να παρέχει την επιλογή αναπαραγωγής ενός τοπικού αρχείου μουσικής. Αυτή η επιλογή μπορεί να σας βοηθήσει να εξοικειωθείτε με τον player και στο debug.

4.1 Σύστημα αναπαραγωγής πολυμέσων

Ο client χρησιμοποιεί το GStreamer για την αναπαραγωγή των πολυμέσων. Οι απαραίτητες ενέργειες που απαιτούνται για την ρύθμιση του GStreamer σας παρέχονται υλοποιημένες. Αυτό που σας ζητείται είναι να λάβετε τα πακέτα και να τροφοδοτήσετε το σύστημα αναπαραγωγής. Για να το πετύχετε αυτό θα πρέπει να επεκτείνετε την συνάρτηση *feed_data_network()* που βρίσκεται στο αρχείο */media/mediaplayer.c*. Στο ίδιο αρχείο μπορεί να σας βοηθήσει η συνάρτηση *feed_data_local()* η οποία είναι πλήρως λειτουργική και μπορεί και αναπαράγει μουσικά αρχεία που βρίσκονται σε κάποιον τοπικό δίσκο.

Και οι δύο συναρτήσεις παίρνουν ως παράμετρο ένα struct τύπου *stream_info_t* (αρχείο */media/mediaplayer.h*) το οποίο χρειάζεται να επεκτείνετε για να παρέχετε την κατάλληλη λειτουργικότητα.

Σημείωση: Η αναπαραγωγή ενός stream από το δίκτυο η ενός τοπικού αρχείου θα πρέπει να καθορίζεται μέσω του field *network_stream* του struct *stream_info_t*, ώστε το σύστημα αναπαραγωγής να είναι σε θέση να καλέσει τα κατάλληλα callbacks.

5 Γενικές πληροφορίες

Μπορείτε να κατεβάσετε τον κώδικα του project από εδώ <https://github.com/surligas/MusicOnCloud>

Ο κώδικας κάθε αρχείου περιέχει αρκετά σχόλια που θα σας βοηθήσουν να κατανοήσετε περισσότερο τα ζητούμενα. Για οποιαδήποτε απορία, χρησιμοποιήστε την λίστα του μαθήματος.