

Προηγμένες Δικτυακές Τεχνολογίες

Using sockets

Αιμίλιος Ψαθάς
Βασίλειος Νάστος
Δημήτρης Τζούβας
Μελίνα Παπαδημητρίου

Msc Department of Informatics and Telecommunications
Group project - Ομάδα 1

Κύριος σκοπός της παρουσίασης είναι η πραγματοποίηση, μίας ανάλυσης των μεθοδολογιών δύο άρθρων και εξαγωγή των αντίστοιχων συμπερασμάτων για τα παρακάτω άρθρα. Συγκεκριμένα αναλύονται:

- Socker: Network-application Co-programming with Socket Tracing(Guo-2021, August).
- Socket intents: Leveraging application awareness for multi-access connectivity.(Schmidt-2013, December)

- Η προσέγγιση των Socket βασίζεται στην λογική ενσωμάτωσης δικτύου-εφαρμογών σε επίπεδο υποδοχής Socket με βάση το eBPF που είναι μια επαναστατική τεχνολογία με προέλευση στον πυρήνα του Linux που μπορεί να εκτελεί προγράμματα sand-boxed σε έναν πυρήνα λειτουργικού συστήματος.
- οι προγραμματιστές μπορούν να πραγματοποιήσουν ευέλικτο έλεγχο δρομολόγησης με βάση τη λογική εφαρμογής, καθώς και δυναμική προσαρμογή λογικής εφαρμογής με βάση τις καταστάσεις του δικτύου

Τι προτείνουν οι συγγραφείς

- Δημιουργία του Socker ως μία καινοτόμα εφαρμογή δικτύου με βάση το eBPF
- Λογική λειτουργία εφαρμογής(AF) και λογική λειτουργία ελέγχου(NF) κάτω από το ίδιο περιβάλλον κεντρικού υπολογιστή
- Μεγαλύτερη ασφάλεια πληροφοριών
- Λειτουργία NF ως ενδιάμεσο σημείο για το αντίστοιχο AF για να επικοινωνεί τις πολιτικές δικτύου με το εκτός δίκτυο, διατηρώντας παράλληλα τις λεπτομέρειες εφαρμογής του AF

- **Πολύπλοκη εξάρτηση:** Δυνατότητα δημιουργίας AF με πολλαπλές ροές σε διαφορετικούς χρόνους κάτω από διαφορετικές καταστάσεις, και το NF μπορεί να έχει πολλούς κανόνες που μπορεί να ενεργοποιηθούν σε διαφορετικές εκδηλώσεις
- **Γρήγορος συγχρονισμός:** Χρησιμοποιείται η δυναμική παρακολούθηση ροής

Για την σχεδίαση λύσεων σύνδεσης των AF με τα NF χρησιμοποιήθηκαν οι λύσεις Strawman. Συγκεκριμένα έχουμε:

- Υλοποίηση στο χώρο του χρήστη.
- Υλοποίηση στο χώρο του πυρήνα.

Για την υλοποίηση του συστήματος χρησιμοποιήθηκε συνδυασμός των δύο υλοποιήσεων.

Τα κύρια χαρακτηριστικά του συστήματος Socker είναι:

- Ελεύθερη σύζευξη μεταξύ AF και NF από την άποψη πρωτοκόλου
- Σύνολο προγραμμάτων eBPF
- Επίπεδο δεδομένων Socker
- Χάρτες eBPF
- Προγράμματα eBPF

Για την μοντελοποίηση λειτουργίας του συστήματος Socker χρησιμοποιήθηκε μία ροή εργασιών σύνδεσης TCP με τέσσερις φάσεις λειτουργίας:

- Δημιουργία Socket
- Φάση σύνδεσης
- Φάση Αποστολής
- Κλείσιμο Socket

- Για διαφορετικά επίπεδα συμπίεσης, το Socker πάντα υπερτερεί έναντι του κανονικού προγράμματος υποδοχής και πέτυχε μείωση του χρόνου μεταφόρτωσης κατά 28,5% κατά μέσο όρο
- η ροή μεταφόρτωσης μεγάλων αρχείων χρησιμοποιεί μια διαδρομή με μεγαλύτερο εύρος ζώνης σε σύγκριση με την προεπιλεγμένη διαδρομή.

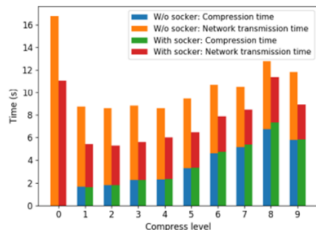


Figure: Αποτελέσματα χρόνου συμπίεσης και μετάδοσης πακέτων για διάφορα επίπεδα συμπίεσης

- Η απόφαση του συστήματος για το ποιο δίκτυο θα χρησιμοποιηθεί βασίζεται στην υποκείμενη δικτυακή υποδομή
- Οι απαιτήσεις της εφαρμογής λαμβάνονται υπόψη όταν αποφασίζεται ποιο δίκτυο θα χρησιμοποιηθεί
- Επιλέγεται η καταλληλότερη διεπαφή δικτύου

Τι προτείνουν οι συγγραφείς

- Ελευθερία στις εφαρμογές να επιλέξουν μόνες τους την μορφή επικοινωνίας που χρειάζονται.
- Βέλτιστο αποτέλεσμα με βάση την επιλογή τους χωρίς κακόβουλες επιπτώσεις στις άλλες εφαρμογές.

- Σχεδιασμός συστήματος που θα καθορίζει **ποιες** διεπαφές θα χρησιμοποιηθούν, **για ποια** επικοινωνία και **με τι** πολιτική.
- Βελτιστοποίηση της διαδικασίας, επιλέγοντας μια από τις εναλλακτικές διαθέσιμες διευθύνσεις προορισμού.

Η υλοποίηση του Socket Intents αποτελείται από τρία στοιχεία:

- Multi Access Manager (MAM)
- τις πολιτικές
- Τη βιβλιοθήκη Socket Intent

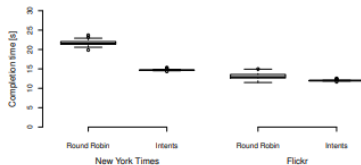
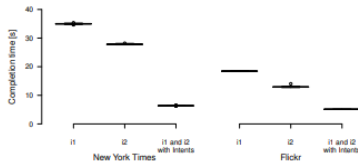
- Ο στόχος του συστήματος Socket Intents είναι να ενεργοποιήσει τις εφαρμογές για να εκφράσουν τις επικοινωνιακές τους προτιμήσεις
- Δεν προορίζονται να καθορίζουν σκληρές απαιτήσεις ή να συνεπάγονται εγγυήσεις QoS
- Είναι εμπνευσμένα από το DiffServ όπου μπορεί, για παράδειγμα, να χρησιμοποιηθεί για την παροχή χαμηλής καθυστέρησης σε κρίσιμη κυκλοφορία δικτύου
- Είναι εμπνευσμένα από το IntServ, καθορίζουν κατηγορίες κυκλοφορίας ανά βάση σύνδεσης

- Μαζική μεταφορά έναντι ερωτήματος
- Αποφόρτωση κατά μέγεθος αρχείου

Πειραματικό μέρος

- Σε μια δίκαιη σύγκριση, η πολιτική "μεγέθους αρχείων" που χρησιμοποιεί ένα κατώτατο όριο συγκρίθηκε με μια πολιτική που δεν γνώριζε η εφαρμογή ένα κατώτατο όριο συγκρίθηκε με μια πολιτική που δεν γνώριζε η εφαρμογή και η οποία εναλλάσσεται μεταξύ δύο διεπαφών για τη λήψη κάθε αντικειμένου.
- Η ρύθμιση έγινε πιο διαφοροποιημένη με τη μείωση του εύρους ζώνης μιας από τις διεπαφές για την προσομοίωση πιο αργών γραμμών DSL.

Το πείραμα δείχνει ότι ο δικτυακός τόπος των New York Times επωφελείται σημαντικά από τη χρήση της πρόθεσης "filesize" με το χρόνο λήψης να βελτιώνεται κατά 35%.



- Το συμπέρασμα του πρώτου άρθρου, είναι ότι η προτεινόμενη προσέγγιση του Socker μπορεί να επιτύχει αποτελεσματικά τον από κοινού έλεγχο των εφαρμογών και του δικτύου μέσω της χρήσης των socket intents και κοινού δίαυλου δεδομένων.
- Το συμπέρασμα του δεύτερου άρθρου, είναι ότι η προτεινόμενη προσέγγιση των Socket intents είναι ένας πολλά υποσχόμενος τρόπος αξιοποίησης της επίγνωσης των εφαρμογών για τη βελτίωση του χειρισμού της συνδεσιμότητας πολλαπλής πρόσβασης.
- Τα δύο άρθρα παρέχουν σαφείς και συγκεκριμένες προτάσεις για μελλοντική έρευνα. Και οι δύο περιγράφουν με σαφήνεια τους τομείς που πιστεύουν ότι θα επωφεληθούν από περαιτέρω έρευνα και παρέχουν μια σαφή πορεία για τη μελλοντική έρευνα

Τέλος παρουσίασης

Ερωτήσεις? Σχόλια?