

Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○	○○○	○	○○○	○○		○
○○○	○○	○○○○○	○○	○○○○○○○○○		○○
○	○○○○○		○○○			
	○○○		○			
	○○		○○			
			○○○			

ScorpioFS

Επέκταση Κατανεμημένου ομότιμου συστήματος αρχείων

Αντώνης Κουζούπης

Πανεπιστήμιο Πειραιώς
Τμήμα Πληροφορικής

23 Οκτωβρίου 2013



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○	○○○	○	○○○	○○		○
○○○	○○	○○○○○	○○	○○○○○○○○		○○
○	○○○○		○○○			
	○○○		○			
	○○		○○			
	○		○○○			

Περιεχόμενα

Εισαγωγή

Γενικά

Συνιστώσες του ScorpioFS

Σχεδιάγραμμα Δικτύου

Chord

Πληροφορίες για το πρωτόκολλο Chord

Χαρακτηριστικά του Chord

Μηχανισμός δρομολόγησης του Chord

Εισαγωγή κόμβου

Αποχώρηση κόμβου

FUSE

Γενικά για το VFS

Πληροφορίες για το FUSE

ScorpioFS

Αναπαράσταση Αρχείων

Έναρξη & Τερματισμός

Εγγραφή ενός αρχείου

Διαθεσιμότητα Αρχείων

Κονσόλα Διαχείρισης

Ασφάλεια

Μετρήσεις

Γενικά

Πειράματα

Εκτέλεση

Κλείσιμο

Μελλοντική Εργασία

Ευχαριστίες



Τι είναι το ScorpioFS

- Σύστημα αποθήκευσης αντιγράφων ασφαλείας**
 Παρέχει στο χρήστη ένα τοπικά προσαρτημένο σύστημα αρχείων το οποίο αποθηκεύει τα περιεχόμενά του στο δίκτυο.
- Δίκτυο ομότιμων κόμβων**
 Αποτελεί ένα δίκτυο υπολογιστών που παρέχουν στην υπηρεσία μία τοπική αποθήκη καθώς και μία λειτουργία αντιγραφής των αρχείων μεταξύ των κόμβων για την εξασφάλιση της διαθεσιμότητας των δεδομένων.
- Κατανεμημένο σύστημα**
 Το σύστημα αποθήκευσης αρχείων είναι πλήρως αποκεντρωμένο. Όλοι οι κόμβοι στο δίκτυο έχουν ισότιμα δικαιώματα. Κληρονομεί τα πλεονεκτήματα και τα μειονεκτήματα των κατανεμημένων συστημάτων.



Εισαγωγή
●
○○○
○

Chord
○○○
○○
○○
○○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○
○○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Τι είναι το ScorpioFS

- Σύστημα αποθήκευσης αντιγράφων ασφαλείας
Παρέχει στο χρήστη ένα τοπικά προσαρτημένο σύστημα αρχείων το οποίο αποθηκεύει τα περιεχόμενά του στο δίκτυο.
- Δίκτυο ομότιμων κόμβων
Αποτελεί ένα δίκτυο υπολογιστών που παρέχουν στην υπηρεσία μία τοπική αποθήκη καθώς και μία λειτουργία αντιγραφής των αρχείων μεταξύ των κόμβων για την εξασφάλιση της διαθεσιμότητας των δεδομένων.
- Κατανεμημένο σύστημα
Το σύστημα αποθήκευσης αρχείων είναι πλήρως αποκεντρωμένο. Όλοι οι κόμβοι στο δίκτυο έχουν ισότιμα δικαιώματα. Κληρονομεί τα πλεονεκτήματα και τα μειονεκτήματα των κατανεμημένων συστημάτων.



Τι είναι το ScorpioFS

- Σύστημα αποθήκευσης αντιγράφων ασφαλείας
Παρέχει στο χρήστη ένα τοπικά προσαρτημένο σύστημα αρχείων το οποίο αποθηκεύει τα περιεχόμενά του στο δίκτυο.
- Δίκτυο ομότιμων κόμβων
Αποτελεί ένα δίκτυο υπολογιστών που παρέχουν στην υπηρεσία μία τοπική αποθήκη καθώς και μία λειτουργία αντιγραφής των αρχείων μεταξύ των κόμβων για την εξασφάλιση της διαθεσιμότητας των δεδομένων.
- Κατανεμημένο σύστημα
Το σύστημα αποθήκευσης αρχείων είναι πλήρως αποκεντρωμένο. Όλοι οι κόμβοι στο δίκτυο έχουν ισότιμα δικαιώματα. Κληρονομεί τα πλεονεκτήματα και τα μειονεκτήματα των κατανεμημένων συστημάτων.



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○○
○○○
○○

FUSE
○
○○○○

ScorpioFS
○○○
○○
○○
○○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Επέκταση του ScorpioFS

- Βελτίωση διαθεσιμότητας
- Ασφάλεια
- Κονσόλα Διαχείρισης
- Μετρήσεις



Οι συνιστώσες του ScorpioFS Chord

Το μέρος του *ScorpioFS* που υλοποιεί το Chord πρωτόκολλο. Είναι υπεύθυνο για την εύρεση των κόμβων που είναι αποθηκευμένα τα δεδομένα, την εισαγωγή και τη διαγραφή ενός κόμβου από το δίκτυο και τη δρομολόγηση των ερωτημάτων.



Εισαγωγή
○
○●○
○

Chord
○○○
○○
○○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○○
○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Οι συνιστώσες του ScorpioFS Fuse

Υλοποιεί το τοπικό σύστημα αρχείων που αντιλαμβάνεται ο χρήστης. Υλοποιεί τις περισσότερες λειτουργίες ενός συστήματος αρχείων όπως δημιουργία, διαγραφή, επεξεργασία, αντιγραφή κτλ. Χωρίζει μεγάλα αρχεία σε μικρότερα του 1MB και επικοινωνεί με το **Chord** κομμάτι για την αποστολή και αποδοχή δεδομένων.



Εισαγωγή
○
○○●
○

Chord
○○○
○○
○○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○○
○○○
○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

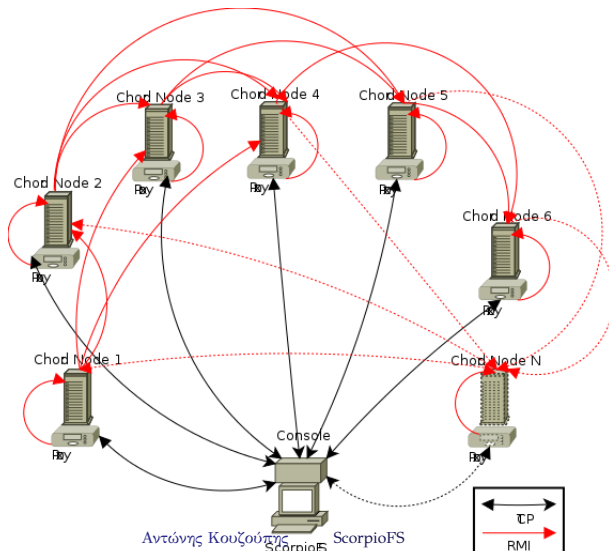
Κλείσιμο
○
○○

Οι συνιστώσες του ScorpioFS Console

Κονσόλα διαχείρισης των κόμβων του δικτύου. Εκτελεί διάφορες λειτουργίες μαζί με τους κόμβους όπως δημιουργία ή καταστροφή, περισυλλογή των στατιστικών. Λειτουργεί ανεξάρτητα από το **Chord** και **Fuse** κομμάτι και επιτελεί επικουρικό ρόλο στο σύστημα.



Σχεδιάγραμμα Δικτύου



Το πρωτόκολλο Chord

- University of California, Berkeley & MIT Laboratory for Computer Science – SIGCOMM'01
- Επεκτάσιμο πρωτόκολλο για αναζήτηση σε ένα δυναμικό peer-to-peer σύστημα με συχνές αφίξεις και αναχωρήσεις κόμβων.
- Τα κλειδιά των κόμβων παράγονται με την εφαρμογή της SHA-1 συνάρτησης κατακερματισμού στην IP διεύθυνση των κόμβων και της θύρας που τρέχει η υπηρεσία.
- Τα κλειδιά των δεδομένων με την εφαρμογή της SHA-1 συνάρτησης στο περιεχόμενό τους.



Το πρωτόκολλο Chord

- Δοθέντος ενός κλειδιού το αντιστοιχίζει σε ένα κόμβο.
- *Consistent hashing* για εξισορρόπηση του φόρτου εργασίας, κάθε κόμβος είναι υπεύθυνος για περίπου τον ίδιο αριθμό κλειδιών, ελάχιστες μετακινήσεις κλειδιών όταν ένας κόμβος μπαίνει ή βγαίνει από το σύστημα.
- Σε ένα σύστημα με N κόμβους, κάθε κόμβος κρατάει πληροφορία για μόνο $O(\log N)$ άλλους κόμβους.



Το πρωτόκολλο Chord

- Επιλύει όλες τις αναζητήσεις μέσω $O(\log N)$ μηνυμάτων προς άλλους κόμβους.
- Το πρωτόκολλο παρέχει μία *lookup(key)* λειτουργία που βρίσκει την IP διεύθυνση του κόμβου που είναι υπεύθυνος για το κλειδί.
- Το Chord ενημερώνει τους κόμβους για τις αλλαγές των κλειδιών που είναι υπεύθυνοι.



Χαρακτηριστικά του Chord

- **Load balance** – Το Chord λειτουργεί σαν κατανεμημένη συνάρτηση κατακερματισμού διαμοιράζοντας τα κλειδιά σε όλους τους κόμβους.
- **Decentralization** – Κανένας κόμβος δεν είναι πιο σημαντικός από τους άλλους. Κατάλληλο για χαλαρά συνδεδεμένες peer-to-peer εφαρμογές.
- **Scalability** – Το κόστος μιας αναζήτησης αυξάνεται λογαριθμικά σε σχέση με το πλήθος των κόμβων.
- **Availability** – Ρυθμίζει αυτόματα το δίκτυο ώστε να “κρύψει” από την εφαρμογή τις αποχωρήσεις και τις αφίξεις νέων κόμβων.
- **Flexible naming** – Δεν θέτει κάποιο περιορισμό στη μορφή των κλειδιών.



Virtual File System

- Αφαιρετικό επίπεδο πάνω από ένα πιο συμπαγές σύστημα αρχείων.
- Επιτρέπει σε εφαρμογές χρήστη (user space) να έχουν πρόσβαση σε ένα συμπαγές σύστημα αρχείων.
- Διαφανής χρήση αποθηκευτικών μέσων χωρίς ο χρήστης να καταλαβαίνει τη διαφορά.
- Το VFS προσθέτει μία διεπαφή μεταξύ του πυρήνα και του συμπαγούς συστήματος αρχείων. Συνεπώς είναι εύκολη η δημιουργία νέων συστημάτων αρχείων.



Filesystem in USErspace

- Μέρος του προγράμματος A Virtual Filesystem (AVFS), αλλά τώρα είναι ανεξάρτητο.
- Ανοιχτό λογισμικό με άδεια χρήσης GNU GPL και GNU Library GPL.
- Διαθέσιμο για Unix-like λειτουργικά συστήματα, Linux, FreeBSD, NetBSD, Mac OS X, OpenSolaris, GNU/Hurd.
- Επίσημα στον πυρήνα του Linux από την έκδοση 2.6.14



Filesystem in USErspace

- Unix kernel module που επιτρέπει σε μη προνομιούχους χρήστες να δημιουργήσουν το δικό τους σύστημα αρχείων.
- Είναι μία “γέφυρα” μεταξύ του πυρήνα και της εφαρμογής του χρήστη.
- Χρησιμοποιεί inode cache και data buffers για βελτίωση στην ταχύτητα.
- Απλό API με bindings σε πολλές γλώσσες προγραμματισμού.
- Απλή εγκατάσταση χωρίς την μεταγλώττιση του πυρήνα.
- Ανάπτυξη με γνώμονα την ασφάλεια.
- Σταθερό!



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○
○○○
○○

FUSE
○
○○●○○

ScorpioFS
○○○
○○
○○○
○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Filesystem in USErspace

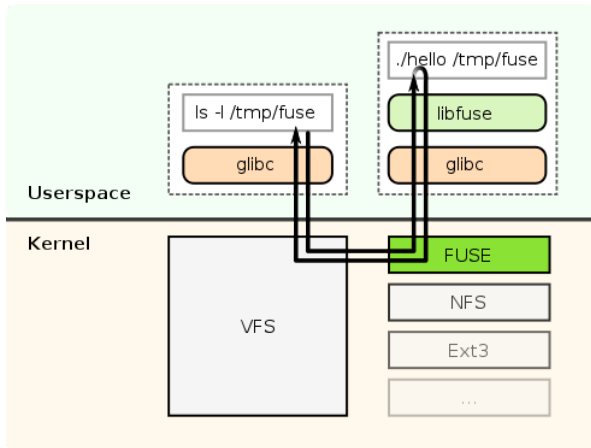
Εφαρμογές που χρησιμοποιούν το FUSE:

- Wuala
- SSHFS
- NTFS-3G
- TrueCrypt
- vmware-mount
- ...



Filesystem in USErspace

Η δομή του FUSE:



FUSE-J

- Java API που παρέχει στο χρήστη bindings για το FUSE.
- Χρησιμοποιεί το framework Java Native Interface (JNI), που επιτρέπει σε κλάσεις Java να καλέσουν ή να καλεστούν (από) native προγράμματα γραμμένα σε γλώσσες όπως C, C++, assembly.
- Ανοιχτό λογισμικό με άδεια GNU Library GPL.



Αναπαράσταση Αρχείων

- Τα αρχεία χωρίζονται σε chunks μεγέθους 1MB.
- Τα δεδομένα είναι σε μορφή πίνακα από bytes.
- Κάθε chunk υλοποιείται από ένα instance της κλάσης *DataObject*.
- Όλα τα chunks – *DataObject* που αποτελούν ένα αρχείο κρατούνται στην κλάση *DataList*. Κάθε αρχείο έχει ένα instance αυτής της κλάσης.



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
●●○
○○
○○○
○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

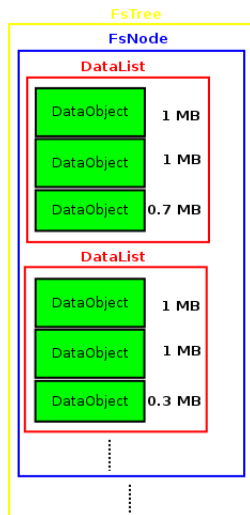
Κλείσιμο
○
○○

Αναπαράσταση Αρχείων

- Κάθε “κόμβος” στο ScorpioFS αποτελεί ένα instance της κλάσης *FsNode*.
- Κρατάει διάφορες πληροφορίες για έναν “κόμβο” όπως όνομα, μέγεθος, τον πατέρα του κόμβου, *DataList*, κτλ
- Όλοι οι κόμβοι – *FsNode* s κρατούνται στην κλάση *FsTree*.
- Το *FsTree* είναι μία συλλογή από *FsNode* s. Επομένως η κλάση *FsTree* υλοποιεί το σύστημα αρχείων.
- Η κλάση *FsTree* είναι serializable.



Αναπαράσταση Αρχείων



Έναρξη ScorpioFS

- Υπάρχει τοπικά ένα αρχείο το οποίο περιέχει τα keys από τα chunks από τα οποία αποτελείται το *FsTree*.
- Το αρχείο αυτό είναι κρυπτογραφημένο με AES-128 αλγόριθμο κρυπτογράφησης και PBKDF2 (Password-Based Key Derivation Function 2) συνάρτηση παραγωγής κλειδιού.
- Αρχικά η εφαρμογή αποκρυπτογραφεί το αρχείο και ανακτά από το δίκτυο τα chunks του *FsTree*.
- Κάνει deserialize το αρχείο, προσαρτάται ένας κατάλογος και “χτίζει” εκεί το σύστημα αρχείων.



Εισαγωγή ○ ○○○ ○	Chord ○○○ ○○ ○○○ ○○○ ○○○ ○○	FUSE ○ ○○○○○	ScorpioFS ○○○ ○○ ● ○○○ ○ ○ ○○ ○○○	Μετρήσεις ○○ ○○○○○○○○	Εκτέλεση	Κλείσιμο ○ ○○
---------------------------	---	--------------------	---	-----------------------------	----------	---------------------

Τερματισμός ScorpioFS

- Γίνεται serialize το *FsTree*.
- Εάν είναι μεγαλύτερο από 1 MB, σπάει σε μικρότερα chunks και στέλνονται στο δίκτυο όπως τα κανονικά αρχεία.
- Αποθηκεύονται τα keys των παραπάνω chunks σε ένα αρχείο.
- Κρυπτογραφείται το αρχείο και αποθηκεύεται τοπικά.
- Αποπροσαρτάται ο τοπικός κατάλογος και σταματάει το πρόγραμμα.



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○
○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○
●○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Εγγραφή ενός αρχείου

- Κλήση της κατάλληλης μεθόδου που επικοινωνεί με το FUSE για την εγγραφή του αρχείου στο σύστημα αρχείων.
- Δημιουργία ενός νέου *FsNode* και αρχικοποίησή του.
- Δημιουργία *DataObject* s με μέγεθος 1 MB.
- Δημιουργία *DataList* με τα παραπάνω *DataObject* s.
- Προσάρτηση του τρέχοντα κόμβου – *FsNode* στο κατάλληλο σημείο του *FsTree*.
- Αφού κλείσει ο file descriptor για το συγκεκριμένο αρχείο, τότε ξεκινάει η διαδικασία για την αποστολή του στο δίκτυο.



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○
○○●○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Εγγραφή ενός αρχείου

- Δημιουργείται το SHA-1 hash του *DataObject*.
- Ο κόμβος συμβουλεύεται το Finger Table του και σύμφωνα με το παραπάνω hash αποφασίζει ποιος κόμβος είναι υπεύθυνος για αυτό.
- Επικοινωνεί με τον κόμβο αυτό με RMI κλήση και του στέλνει το chunk.



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○ ○○○ ○	○○○ ○○ ○○○○ ○○○ ○○	○ ○○○○○	○○○ ○○ ○○● ○ ○○ ○○○	○○ ○○○○○○○○		○ ○ ○○

Ανάγνωση ενός αρχείου

- Καλείται η *read* μέθοδος από το FUSE.
- Αν το συγκεκριμένο *DataObject* υπάρχει στην τοπική αποθήκη του κόμβου τότε διαβάζεται από εκεί.
- Διαφορετικά μέσω της λειτουργίας που ορίζει το Chord πρωτόκολλο, ο κόμβος βρίσκει ποιος κόμβος είναι υπεύθυνος για το chunk αυτό.
- Μέσω RMI κλήσης φέρνει το chunk και το ανοίγει.
- Υπάρχει η δυνατότητα για την ανάκτηση συγκεκριμένων chunks και όχι όλου του αρχείου.



Αντιγραφή Αρχείων

- Ανά τακτά χρονικά διαστήματα – 5 δευτερόλεπτα – ξεκινάει η διαδικασία αντιγραφής αρχείων μεταξύ των κόμβων στο δίκτυο.
- Για κάθε ένα αρχείο που έχει στην αποθήκη του ο κόμβος, υπολογίζεται πάλι το SHA-1 hash του.
- Αναζητείται ο υπεύθυνος κόμβος για το νέο πλέον key και μέσω RMI κλήσης αντιγράφεται σε αυτόν.
- Αν ο νέος υπεύθυνος κόμβος δεν είναι γνωστός για τον τρέχοντα τότε ψάχνει αναδρομικά.
- Η παραπάνω διαδικασία περιορίζεται από ένα Replication Factor.



Εισαγωγή ○ ○○○ ○	Chord ○○○ ○○ ○○○○ ○○○ ○○	FUSE ○ ○○○○○	ScorpioFS ○○○ ○○ ○○○ ○ ○○ ○○○	Μετρήσεις ○○ ○○○○○○○○	Εκτέλεση	Κλείσιμο ○ ○○
---------------------------	---	--------------------	---	-----------------------------	----------	---------------------

Διαθεσιμότητα metadata

- **Κεντρικό** αρχείο–κλάση που αποθηκεύει τη δενδρική δομή του συστήματος αρχείων.
- Το ScorpioFS διάβαζε από τον τοπικό δίσκο την κλάση και δομούσε πάλι τα περιεχόμενα του mount point.
- Αδυναμία ανάκτησης των αρχείων σε περίπτωση απώλειας ή μερικής καταστροφής του.



Διαθεσιμότητα metadata

- Serialize της κλάσης που αποθηκεύει τη δομή του συστήματος αρχείων.
- Τυχών διαχωρισμός σε περισσότερα του ενός αρχεία.
- Μέσω της συνάρτησης κατακερματισμού βρίσκουμε το id των αρχείων.
- Βρίσκουμε τους successors των παραπάνω αρχείων σύμφωνα με τη *lookup* μέθοδο του Chord.
- Αποστολή των αρχείων στους successors.
- Κάθε κόμβος ανά τακτά χρονικά διαστήματα δημιουργεί καινούργιο id εφαρμόζοντας πάλι hash function και βρίσκει τους νέους successors.
- Το Chord μας “εξασφαλίζει” την ανάκτηση του αρχείου.



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○ ○○○ ○	○○○ ○○ ○○○○ ○○○ ○○	○ ○○○○○	○○○ ○○ ○○○ ○ ○ ●○ ○○○	○○ ○○○○○○○○		○ ○ ○○

Κονσόλα Διαχείρισης

- Έχει επικουρικό ρόλο στο σύστημα. Βοηθάει για τη μαζική εκτέλεση ενεργειών στους κόμβους.
- Αρχιτεκτονική πελάτη–εξυπηρετητή χρησιμοποιώντας το TCP πρωτόκολλο επικοινωνίας.
- Η κονσόλα διαχείρισης στέλνει μηνύματα στους κόμβους.
- Σε κάθε Chord κόμβο υπάρχει ένας proxy εξυπηρετητής που λαμβάνει τα μηνύματα και τα προωθεί.
- Στην κονσόλα διαχείρισης υπάρχει ένας πολυνηματικός εξυπηρετητής που λαμβάνει τα μηνύματα των κόμβων.



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○○
○
○
●
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Κονσόλα Διαχείρισης

Οι λειτουργίες που προσφέρει η κονσόλα διαχείρισης είναι:

- Δημιουργία κόμβων μεμονωμένα ή μαζικά.
- Τερματισμός κόμβων μεμονωμένα ή μαζικά.
- Απαρίθμηση των ενεργών κόμβων στο δίκτυο.
- Συγκέντρωση και εξαγωγή σε αρχεία διάφορων μετρήσεων από τους κόμβους.



Κονσόλα Διαχείρισης

- **ConsoleServer** Αυτόνομο νήμα που υλοποιεί τον proxy server σε κάθε κόμβο
- **ConsoleClient** Κονσόλα διαχείρισης κόμβων μαζικά, επικοινωνεί με κάθε κόμβο με συγκεκριμένο πρωτόκολλο
- **ConsoleClientReceiver** Αυτόνομο νήμα που το ξεκινάει η κονσόλα διαχείρισης με το οποίο επικοινωνούν οι κόμβοι για να στείλουν μηνύματα επιβεβαίωσης



Εισαγωγή	Chord	FUSE	ScorioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○	○○○	○	○○○	○○		○
○○○	○○	○○○○○	○○	○○○○○○○○○		○○
○	○○○○○		○○○			
	○○○		○			
	○○		○○			
			○○○			

Λειτουργίες της κονσόλας

- **node create IP_ADDR[:PORT] [-chordport PORT -config CONFIG]**
node create 10.1.210.4:6789 -chordport 6782 -config config/chord.config
- **node create -f FILE**
node create -f chordNodes
#Proxy IP,Proxy port,chord node port,configuration file
- **node stop IP_ADDR[:PORT] [-chordport PORT]**
node stop 10.1.210.4:6789 -chordport 6782
- **node stop -f FILE**
node stop -f chordNodes



Εισαγωγή
○
○○○
○

Chord
○○○
○○
○○○○
○○○
○○

FUSE
○
○○○○○

ScorpioFS
○○○
○○
○○
○○○
○
○○
○○○

Μετρήσεις
○○
○○○○○○○○

Εκτέλεση

Κλείσιμο
○
○○

Λειτουργίες της κονσόλας

- **node stats get**
- **node stats export**
- **node stats clear**
- **node list**



Ασφάλεια στο ScorpioFS

- Η κρυπτογράφηση κάθε αρχείου μεμονωμένα δεν είναι αποδοτική
- Για την αναπαράσταση του αρχείου συστήματος υποχρεωτικά θα πρέπει να υπάρχουν τα κατάλληλα metadata.
- Πρόσβαση στα αρχεία σε περίπτωση που κάποιος κακόβουλος χρήστης του δικτύου έχει πρόσβαση σε αυτά.
- Κρυπτογράφηση των metadata!



Ασφάλεια στο ScorpioFS

- The Legion of Bouncy Castle cryptographic provider
- Advanced Encryption Standard symmetric encryption
- Cipher Block Chaining (CBC)
- Password-Based Key Derivation Function 2 128 bits
- HMAC-SHA-1 pseudorandom function



Ασφάλεια στο ScorpioFS

- Κρυπτογράφηση των metadata που δομούν το σύστημα αρχείων
- Αποστολή στο δίκτυο
- Κρυπτογράφηση του τοπικού φακέλου που περιέχει τα κλειδιά που αντιστοιχούν στα metadata καθώς και τα Initialization Vectors και Salt που χρησιμοποιήθηκαν κατά την κρυπτογράφηση
- Συμπίεση του αρχείου με τα κλειδιά που αντιστοιχούν στα chunks του αρχείου με τα metadata, του Initialization Vector και του Salt σε ένα zip
- Κρυπτογράφηση του παραπάνω zip αρχείου
- Η τοποθεσία που αποθηκεύεται καθορίζεται από το χρήστη μέσω ενός αρχείου παραμετροποίησης



Μετρήσεις

Οι μετρήσεις που παίρνουμε είναι οι παρακάτω:

- Χρονική διάρκεια που λειτουργεί ο κόμβος
- Αιτήσεις Put
- Αιτήσεις Get
- Συνολικό αριθμό από chunks που εξυπηρετεί ο κόμβος
- Χώρο που καταλαμβάνουν τα παραπάνω chunks



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○ ○○○ ○	○○○ ○○ ○○○ ○○○ ○○	○ ○○○○○	○○○ ○○ ○○○ ○○○ ○ ○○ ○○○	○● ○○○○○○○○		○ ○○

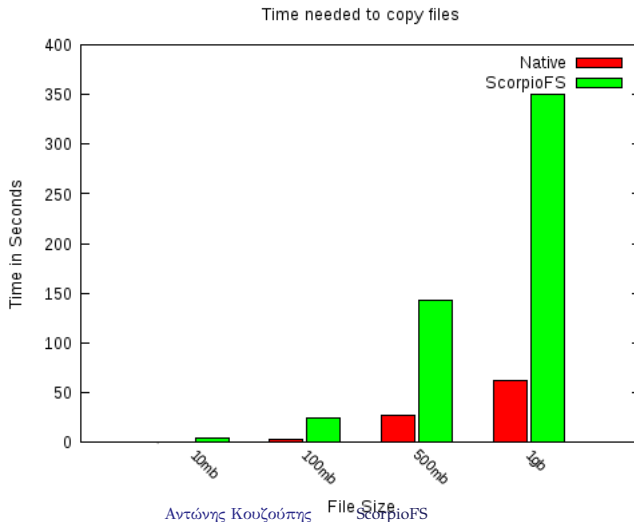
Testbed

Χώρος διεξαγωγής των μετρήσεων ήταν τα εργαστήρια του τμήματος

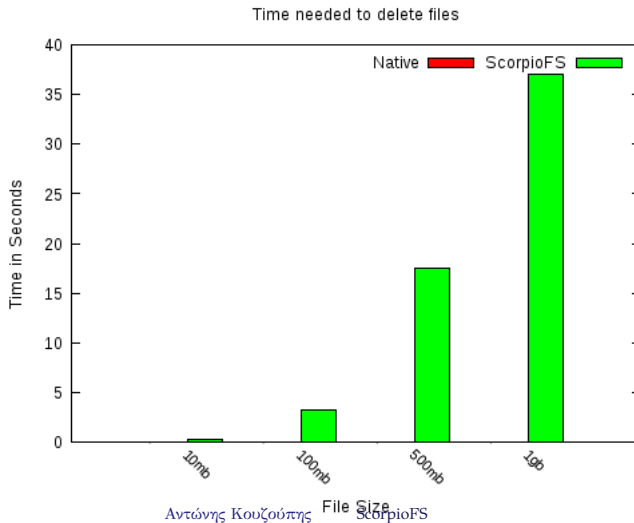
- 20 φυσικοί κόμβοι
- 58 chord κόμβοι
- Debian GNU/Linux 6.0
- 1 GB RAM
- 1 GbE



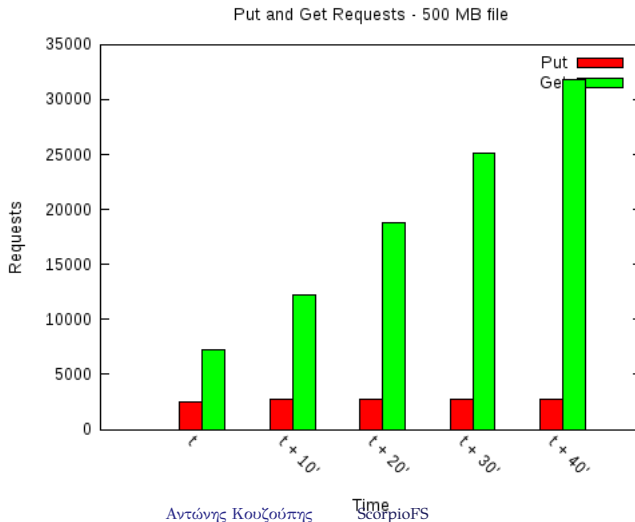
Αντιγραφή Αρχείων



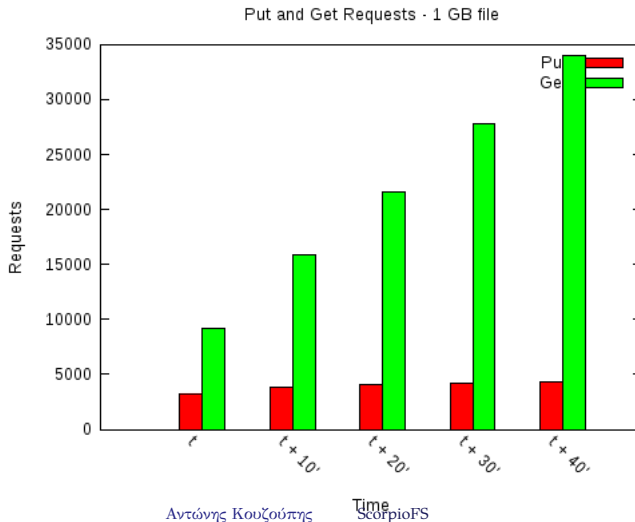
Διαγραφή Αρχείων



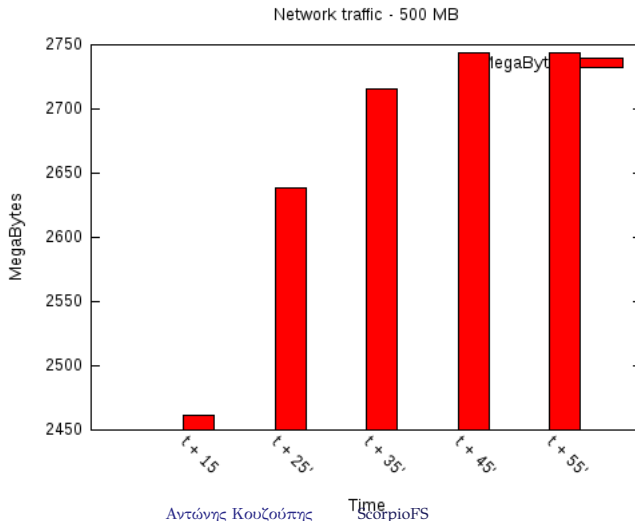
Put & Get Requests – 500 MB file



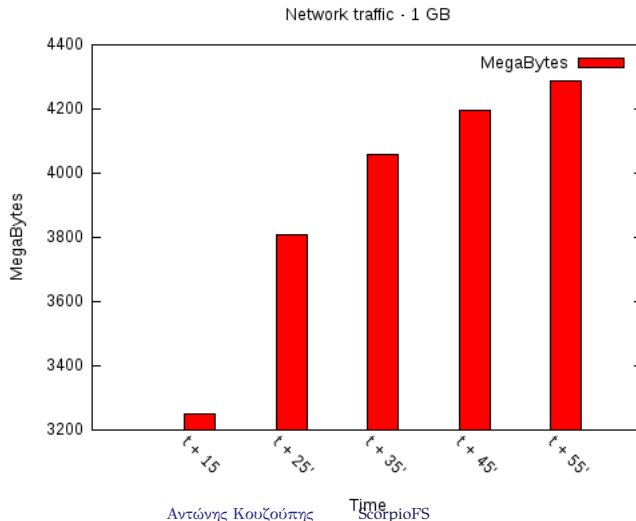
Put & Get Requests – 1 GB file



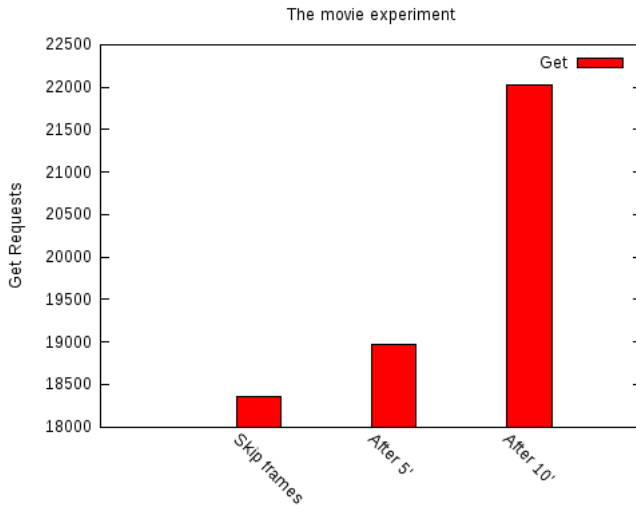
Network Traffic – 500 MB file



Network Traffic – 1 GB file



Get Requests – Τυχαία κλήση chunks



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○ ○○○ ○	○○○ ○○ ○○○○ ○○○ ○○	○ ○○○○○	○○○ ○○ ○○○ ○ ○○ ○○○	○○ ○○○○○○○●		○ ○○

Διαθεσιμότητα

Το περιεχόμενο της ταινίας ήταν διαθέσιμο μέχρι και όταν $52/58 \simeq \log(58)$ κόμβους του δικτύου είχαν τερματίσει τη λειτουργία τους αφού “έτρεχαν για 30”

Τα metadata που διαμορφώνουν τη δενδρική μορφή του συστήματος αρχείων διαμοιράζονται στο δίκτυο για μεγαλύτερη διαθεσιμότητα.



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○ ○ ○○○ ○	○○○ ○○ ○○○ ○○○ ○○	○ ○○○○○	○○○ ○○ ○○○ ○ ○○ ○○○	○○ ○○○○○○○○		○ ○ ○○

Διαθεσιμότητα

```

antonis@wizard: ~/scorpiofs/java$ ./scorpio.sh -port 6788 -config config/scorpiofs.properties
~/sco/
[Password:]
  decode secret key is:8f7a52995228f39f5513ce719a017346
  decode secret key 2 is:8f7a52995228f39f5513ce719a017346
Extracting file: /home/antonis/.scorpiofs/interFs
Extracting file: /home/antonis/.scorpiofs/.iv
Extracting file: /home/antonis/.scorpiofs/.salt
Checksum: 720635344
20:24:44.393      main  INFO [fuse.scorpiofs.ScorpioFS]: Just before revoking fstree chunks
20:24:44.395      main  INFO [fuse.scorpiofs.util.RetrieveChunks]: interFile exists: true35
8
20:24:44.403      main  INFO [fuse.scorpiofs.util.RetrieveChunks]: Chunk id: 16480730296840
5432161255653061503146651980724460
20:24:44.405      main  INFO [fuse.scorpiofs.util.ChunkNetwork]: Chunk id: 1648073029684054
32161255653061503146651980724460
20:24:44.425      main  INFO [fuse.scorpiofs.util.ChunkNetwork]: fstree chunk found, first o
: 10.1.210.14
20:24:44.426      main  INFO [fuse.scorpiofs.util.RetrieveChunks]: size of buffer: 31120
decode secret key is:1327d6c8b4ca06a93634e2f58cc2d081

```



Εισαγωγή	Chord	FUSE	ScorpioFS	Μετρήσεις	Εκτέλεση	Κλείσιμο
○ ○ ○○○ ○	○○○ ○○ ○○○○ ○○○ ○○	○ ○○○○○	○○○ ○○ ○○○ ○ ○○ ○○○	○○ ○○○○○○○○		○ ○ ○○

Διαθεσιμότητα

```

antonis@wizard: ~/scorpiofs/java$ ./scorpio.sh -port 6788 -config config/scorpiofs.properties
~/sco/
[Password:]
  decode secret key is:8f7a52995228f39f5513ce719a017346
  decode secret key 2 is:8f7a52995228f39f5513ce719a017346
Extracting file: /home/antonis/.scorpiofs/interFs
Extracting file: /home/antonis/.scorpiofs/.iv
Extracting file: /home/antonis/.scorpiofs/.salt
Checksum: 720635344
20:24:44.393      main  INFO [fuse.scorpiofs.ScorpioFS]: Just before revoking fstree chunks
20:24:44.395      main  INFO [fuse.scorpiofs.util.RetrieveChunks]: interFile exists: true35
8
20:24:44.403      main  INFO [fuse.scorpiofs.util.RetrieveChunks]: Chunk id: 16480730296840
5432161255653061503146651980724460
20:24:44.405      main  INFO [fuse.scorpiofs.util.ChunkNetwork]: Chunk id: 1648073029684054
32161255653061503146651980724460
20:24:44.425      main  INFO [fuse.scorpiofs.util.ChunkNetwork]: fstree chunk found, first o
n 10.1.210.3
20:24:44.486      main  INFO [fuse.scorpiofs.util.RetrieveChunks]: size of buffer: 31120
  decode secret key is:1327d6c8b4ca06a93634e2f58cc2d081

```



Εκτέλεση

Για να δημιουργήσουμε την υποδομή για το ScorpioFS θα πρέπει:

1. Σε κάθε κόμβο να υπάρχει το εκτελέσιμο της εφαρμογής
2. Στον κόμβο που θα προσαρτηθεί το ScorpioFS filesystem να υπάρχει υποστήριξη για το FUSE
3. Προσαρμογή των αρχείων παραμετροποίησης για το Chord σύστημα (ip διεύθυνση, πόρτα λειτουργίας, bootstrap κόμβος, κτλ) και για το ScorpioFS (τοποθεσία αποθήκευσης αρχείων συστήματος)
4. Έναρξη του proxy server σε κάθε κόμβο
5. Έναρξη της κονσόλας διαχείρισης και δημιουργία των κόμβων
6. Εκτέλεση του ScorpioFS προγράμματος που προσαρτά το σύστημα αρχείων
7. Αντίστροφη διαδικασία για το τερματισμό της λειτουργίας του συστήματος



Μελλοντική Εργασία

- Αυθεντικοποίηση και πιστοποίηση των κόμβων
- Διαγραφή αρχείων από τους κόμβους
- Δημιουργία χρηστών και usergroups
- NAT traversal
- Περιορισμός του χώρου αποθήκευσης σε κάθε κόμβο
- GUI



Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά:

- Καθηγητή κ. Δουληγέρη Χρήστο
- Δρ. κ. Αβραμίδα Αγάπιο
- Εργαστήριο Τμ. Πληροφορικής



Εισαγωγή

○
○○○
○

Chord

○○○
○○
○○○○
○○○
○○

FUSE

○
○○○○○

ScorpioFS

○○○
○○
○○○
○
○○
○○○

Μετρήσεις

○○
○○○○○○○○

Εκτέλεση

Κλείσιμο

○
○●

Q&A

```
antonis@wizard:~/ScorpioFs/java$ ./console.sh
Welcome to ScorpioFS administration console
Type help for more info
$>finito
```



```
+-----+
|Antonis kouzoupis - P/06073|
|kouzoupis.ant@gmail.com |
+-----+
```

Thank you!!!

\$>|

