Κατανεμημένα Συστήματα, NTUA 2019 - 2020

Αναφορά εξαμηνιαίας εργασίας

Σύντομη περιγραφή:

Η παρούσα αναφορά αφορά τη δημιουργία ενός απλού συστήματος blockchain, όπου καταγράφονται οι δοσοληψίες μεταξύ των συμμετεχόντων και εξασφαλίζεται το consensus με proof-of-work.

Για την ανάπτυξη της εφαρμογής επιλέχθηκε η γλώσσα Python και έγινε χρήση του flask api για τη δημιουργία του REST (για ανταλλαγή μηνυμάτων, αιτήσεις χρηστών προς το σύστημα, είσοδος νέου κόμβου στο σύστημα κ.λ.π.)

Το σύστημα "τρέχει" σε cluster 5 VMs στον okeanos.

**ADD ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ ΕΔΩ**

Client

Για τις ανάγκες της εφαρμογής έχουμε δημιουργήσει ένα απλό cli (cli.py) που δίνει στο χρήστη τη δυνατότητα να δημιουρήσει κάποια συναλλαγή, να δει τις πιο πρόσφατες συναλλαγές του συστήματος, να δει το προσωπικό του υπόλοιπο NBCs και τέλος πληκτρολογώντας help, μπορεί να δεί λεπτομέρειες για τις παραπάνω λειτουργίες.

Πειράματα

1) Απόδοση

Εισέρχονται ***5*** clients στο σύστημα και ο καθένας (Χ) διαβάζει από ένα αρχείο (transactionsX.txt) τα transaction τα οποία πρόκειται να εκτελέσει.

Γίνονται δοκιμές για διαφορετικές τιμές χωρητικότητας των block (1,5,10) & difficulty (4,5):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **capacity, difficulty** | **1,4** | **1,5** | **5,4** | **5,5** | **10,4** | **10,5** |
| **throughput** |  |  |  |  |  |  |
| **block time** |  |  |  |  |  |  |

\*throughput: transactions που εξυπηρετούνται/χρόνο

\*block time: μέσος χρόνος για να προστεθεί νέο block στο chain

2) Κλιμακωσιμότητα

Τα παραπάνω επαναλαμβάνονται για ***10*** clients:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **capacity, difficulty** | **1,4** | **1,5** | **5,4** | **5,5** | **10,4** | **10,5** |
| **throughput** |  |  |  |  |  |  |
| **block time** |  |  |  |  |  |  |

+2 bar plots: 1. για throughput (y), για κάθε συνδυασμό capacity, difficulty 2 bars (1 για 5, 1 για 10 clients. 2. Τα ίδια για block time.