Distributed Systems (CS3.401)
Spring 2022
Project Report

Project Title: Distributed Password Cracker

Team Members (Team #5)

Akshit Garg (2018113006)
Amogh Tiwari (2018111003)
Utkarsh Mishra (2018102020)

Problem Statement

Distributed Password Cracker: A password cracker using brute force but takes advantage of multiple, networked computers

Introduction

Our project gives a demo of a distributed password cracker having one server and multiple clients. To use the system, the user first generates a hash for their password using the *hash_gen* file and then the user gives this hash as input to the *server* file. The server distributes the work of cracking the password among all the clients by distributing the whole search space among different clients. If one of the clients succesfully cracks the password, then it returns the correct password to the server to be printed.

Algorithm Overview

1. Server gets the hash of the password as a user input. (This models the real life action of server intercepting the password being transferred over the network OR a remote server telling you if the password you entered was correct or not)
2. Server waits for connections to be established with the client(s). Once connection is established, the server gives the clients a subset of the search space to be searched for.
3. The client runs a loop over all possible password combinations which can be generated from the search space range passed to it. It generates a hash of the number and checks if the hash is same as the hash which the server had got
4. If the client doesn't find any match in the search space, it sends a message back to the server asking for a new search space range

5. If two hashes match then the client returns the pattern to the server and the server prints that password. Further, the server then stops broadcasts to all the clients

Constraints/Limitations

1. Number of clients is a system parameter
2. For displaying the progress bar, we assume a maximum password length. However, the max password length parameter doesn't effect the working of the system. It just affects the progress bar
3. This system uses sha5 and md5 hashing functions. But can be easily extended for other types