Abstract

Passwords are one of the commonly used methods to protect one's personal information against the intruders. But storing passwords as plaintext is not safe, hence they are saved in the form of hashes. And authentication occurs by comparing the hash in the database to the hash generated from input taken. It is crucial that the hashing algorithm is not only tough to reverse engineer but, should also be nearly impossible to find a collision [1]. The study considers a different approach using distributed processing to compute multiple hashes at a very high speed, making one of the most widely used hashing algorithm SHA-512[2] seem not that secure after all. The approach involves cryptanalyzing bcrypt, another hashing algorithm, and concluding whether it's a good alternative.

References

1. Lin Zhou and Wenbao Han, "A brief implementation analysis of SHA-1 on FPGAs, GPUs and Cell Processors", 2009 International Conference on Engineering Computation, IEEE, Pages
101-104, May 2009
3. Kelly Brown, “The Dangers of Weak Hashes”, SANS Institute InfoSec Reading Room, June
7. Tatli, E.I., August 2015. “Cracking more password hashes with patterns.” IEEE. Pages 1656-1665

Index Terms

Computer Science

Security

Keywords

Keyspace, Hashing, bcrypt, SHA-512, Cryptanalysis, Distributed processing, Cloud. Cluster