International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 05 | May 2021 www.irjet.net p-ISSN: 2395-0072

Implementing and Comparing Different Password Cracking Tools

Disha Pahuja [1], Prerna Sidana [2]

[1,2]*Students, Computer Science Department, Manav Rachna International Institute of Research and Studies, Sector 43, Faridabad*

Abstract— *This article looks at the essence of the cracking of passwords and modern applications. Several applications are being studied for various platforms. Various cracking methods, including Hashcat, John the Ripper and Fcrackzip, which are being implemented on the Kali Linux Operating System, are explained. Cracking of passwords throughout various media is investigated. Hashing and how password cracking is impacted are discussed. Two hash-based password cracking algorithms are being executed, as well as with experimental results of their effectiveness.*

Keywords— **Passwords, Hashes, Authentication, Brute- force, password-cracking tools**

I. INTRODUCTION

Passwords can be the first choice among all to protect their files, networks, be it their personal computer, email or bank account. The other way malicious users view passwords is the only way to intrude in networks unethically and gain access to confidential information desired to be stolen. These malicious users are authenticated by all the unfair means. The methodology behind can be either malicious scripts or password cracking tools, among other options.

The process of authentication involves a certain number of important steps. The user has to keep a proof of his/her identity i.e., identification. [2] While typing down the login ID and password to log into the computer, it is a username and a password. This will authorise the user to gain control over the system and the confidential data inside. Password security approaches ought to be set up at associations with the goal that staff realize how to make a password, how to store their password and how frequently to transform it. Password assurance is a security cycle that guarantees information accessible through PCs that ought to be protected from explicit customers. Password insurance allows only those with an affirmed password to get to certain information.

II. LITERATURE REVIEW

Passwords, as a type of validation can be supposed to be as old as time. In antiquated occasions, guardians would test those wishing to enter a territory or moving toward it to gracefully a watchword of which if right passageway is given.[8] In present day times nonetheless, a blend of username and password is a typical method for verification during sign in measures. Password cracking instruments can be altogether arranged into disconnected and web-based cracking classes. Attacks, for instance, word reference and brute-force attack are performed against on a live structure login structure or meeting is called online assault .The prevalence of online attacks may not be as much as disconnected attacks on account of how they are commonly unfathomable to pull off as there are different confirmation plans being utilized that can make such an attack problematic and dangerous to recognize anyway it is so far possible to pull off if a part of these frameworks, for instance, most noteworthy inadequate approval attempts.

In this paper, we have first presented prior knowledge about passwords and password cracking. Then we have implemented three password cracking tools namely John the Ripper, Hashcat and Fcrackzip on Kali Linux Platform. We have also presented a Comparative Analysis of the above-mentioned tools based on their functionality and complexity or time taken by them to crack password. The procedure or steps of implementation have also been explained to give an idea about the execution of the tools. Our work focuses on trawling password-guessing attacks.

III. PASSWORD CRACKING

Password cracking is indeed the means to speculate or retrieve a password from remote areas or the information transmission framework. It is used to provide a password unapproved access or to recoup an overlooked password. In entrance checking, it is used to verify the protection of the submission.

Lately, tech developers have been trying to measure the breaking of a password in a shorter span of time. Among others, such applications want and log in for any possible term combination. In the case of a successful authentication, the secret has been found. In the event that a password is adequate with a combination of numbers, characters and unique characters, this cracking technique can take hours to weeks or months.

A few of the cracking systems have a word guide that includes passwords. These methods are totally subject to the complexities of the term, so the rate of achievement is lower.

In the past few years, people in IT industry have developed various cracking tools for passwords. Each of such tools has its own points of interest and obstacles. In this article, we are addressing and examining a few of the most common cracking tools for passwords.

IV. PASSWORD CRACKING TOOLS

The way wherein password breaking ordinarily works is either to pull passwords out of word reference reports or produce them, by then run them through a hash figuring or to question the hashes in a rainbow table and take a gander at them. This fragment intends to cover a part of the more notable password breaking instruments that are used today both by aggressors and structure administrators to test the security of their system .The tools we have executed are as follows:

i. HashCat

ii. John The Ripper

iii. Fcrackzip

V. HASHCAT

Hashcat is the fastest and the best utility for the recovery of passwords. It is a freely available application. Hashcat at present backings CPUs, GPUs, and other equipment quickening agents on Linux, Windows, and OSX, and has offices to help empower circulated password splitting. This utility can convert user-understandable data into a long string of characters but of fixed length, which can't be interpreted by humans. This long string is often referred to as hash.

Certain criteria is followed while generating these hashes. [10] These are better known as hashing algorithms. Some of the popular ones used to generate these long, random, non-interpretable strings are MD5, SHA, WHIRLPOOL, RipeMD, etc. Hashes are commonly defined as a one-way function, the reason being that such generated manipulations are easily manipulated by the algorithms but very difficult to reverse engineer. HashCat is a password cracking application, supporting five unique modes of attack for over 200 highly-optimized hashing algorithms.

HashCat is a password breaking application, supporting five interesting methods of attack for more than 200 exceptionally improved hashing calculations. Hashes doesn't permit to unscramble information with a particular key, as standard encryption conventions allow.Hashcat utilizes precomputed word references, rainbow tables, and even a brute-force way to deal with locate a viable and proficient approach to break passwords.

It has the following features:

- Excellent randomisation of strings
- Operable on multiple OS platforms
- Very fast performance.
- It is multi-Algorithm based.
- The salt-list loaded from the external file can be used as a brute-force attack variant.

V.I. Steps to Implement HashCat

1. **Create a dictionary with MD5 hashes:**
   To begin this demonstration, we will create multiple hash entries, that will be containing several passwords. They are stored to a file named as "pass.txt" The commands for execution are written and executed in the terminal there and then right away, as displayed in the screenshot below:

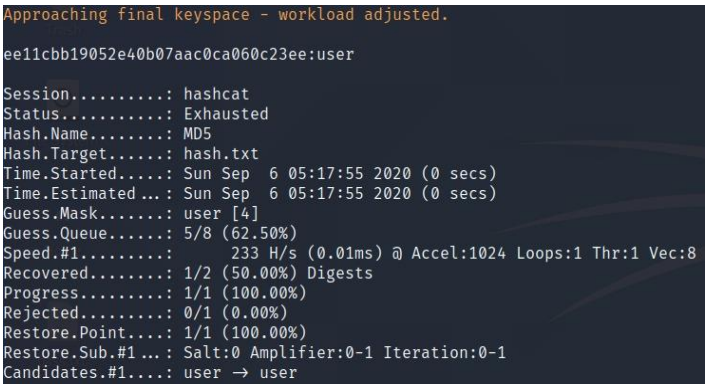

Fig.V.1 Create a dictionary with MD5 Hashes

2. **Start Hash cat in Kali Linux:**

   Hashcat can be started on the Kali console with the following command line:

   hashcat -h.

   Some of the most important hashcat options are :

   - -m (the hashtype)
   - -a (attack mode)

   By and large, we have to utilize the two choices in most password breaking endeavors when utilizing Hashcat. Hashcat likewise has explicitly planned principles to use on a wordlist document. The character rundown can be redone to break the password(s).

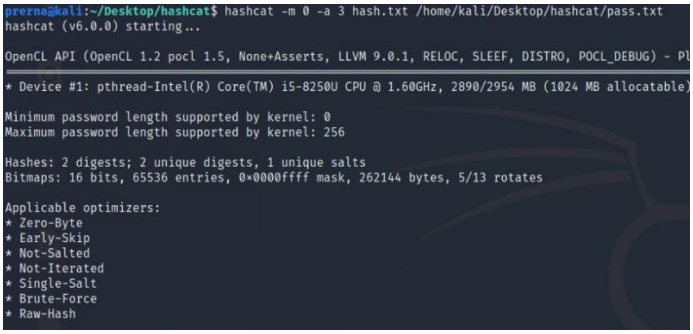

Fig.V.2 Starting HashCat

3. In the final step, we can now start cracking the hashes contained in the hash.txt file. We will use the following command line, as illustrated below:

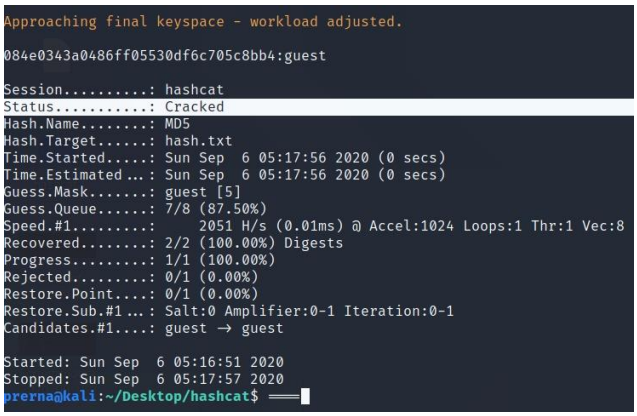

Fig.V.3 Cracking the hashes

VI. JOHN THE RIPPER

John the Ripper is a quick secret phrase wafer which is likewise as of now accessible for a few Unix, MacOS, Windows, DOS, BeOS, and OpenVMS variations. Truly, its essential way to deal with perceive frail Unix passwords. Nowadays, notwithstanding [9] numerous other Unix tomb secret phrase hash types, there have been many extra hashes and ciphertext that are acknowledged in "-large" forms.

It is one of the most normally utilized applications that can test and break passwords as it incorporates an assortment of secret word saltines into one bundle, auto-identifies secret phrase hash frames, and gives an adaptable wafer. It might be run against various encoded mystery key designs, including some grave mystery key hash outlines most normally found in different Unix models (considering DES, MD5, or Blowfish), Kerberos AFS, and Windows NT/2000/XP/2003 LM hash.[7] The ability to use MD4-based mystery word hashes and passwords set aside in LDAP and others has been improved by additional modules.

VI.I. Steps of Implementing John the Ripper To crack the password of Zip And RAR Files:

1) For this we have created password protected RAR and ZIP files, that each contain two files named as “area.zip” and “test.rar”.

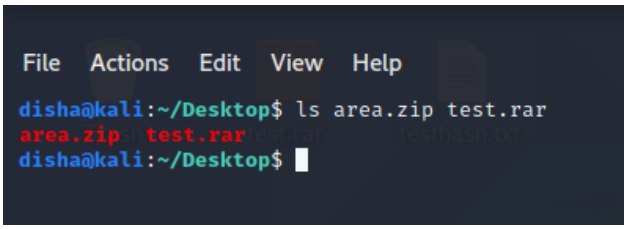

Fig.VI.1 Display password protected files

2) In the 'run' organizer of John the Ripper, there are two projects called 'zip2john' and 'rar2john'. Run them against their particular record types to separate the secret word hashes.

3) Now we need to create a hash for the file that you want to hack. To create the hash and save the hash into a file – Type the command: sudo zip2john area.zip > hash.txt

4) Now write the command sudo john hash.txt to crack the password of the zip file.

Fig. VI.2. Cracking the hash using John the Ripper

5) The same commands are used for rarfile also but instead of zip2john now rar2john is used.

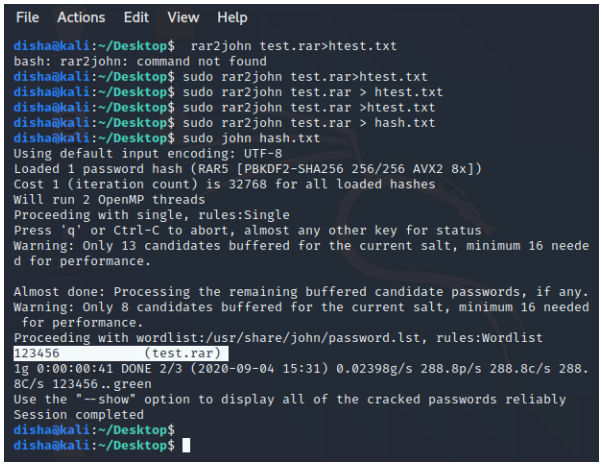

Fig.VI.3. Cracked password for “test.rar” file (highlighted)

VII. FCRACKZIP

As it has a small size and encryption algorithm, we often use zipped files. These zipped files come with a password protection facility that ensures the files' security.

Fcrackzip solves this problem to do this and let you have the way out to safeguard your documents once you have misplaced the password, and the question occurs of how to crack it. A ideal way to unlock a protected zip file that is possible under Linux [11] with the application of fcrackzip.

Fcrackzip was written by Marc Lehmann and is a free / rapid zip password cracker. It was not the quickest zip cracker available, but a versatile zip password cracker that was affordable, but still efficient.

VII.I. Multiple features of Fcrackzip

As we are using Kali Linux, Fcrackzip tool is installed by default, we just need to open the terminal and just type “fcrackzip –help” and its help command will run and mention some important commands, that are frequently to be used while execution of Fcrackzip.

List of Commands:

- -b: for using brute force algorithms.
- -D: for using a dictionary.
- -B: execute a small benchmark.
- -c: use characters from charset.
- -h: show the help message.
- –version: show the version of this program.
- -V: validate or check the algorithm.
- -v: for verbose mode.
- -p: for using a string as a password.
- -l: for providing a specific length to password.

VII.II. Steps To implement Fcrack on a zip file in Kali Linux:

1) Initially, we need to make a password-ensured document in that cycle we have to choose that record which we need to make sure about with that design, in the wake of choosing that document we have to follow the order.

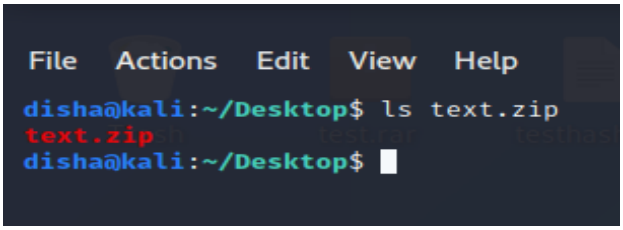

Fig VII.1. Display the test file in terminal

2) Cracking the password of the zip file fcrackzip is an extremely significant apparatus and furthermore very simple to use for making a beast power assault on any compress record, for that we have to utilize diverse distinctive organization for breaking the password of the compress document.In order to that we (-b) which allow us to brute force on that zip file, (-c) which define the charset for the dictionary to brute force.

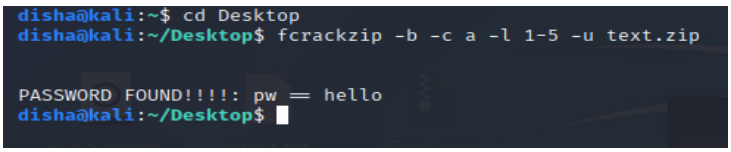

Fig VII.2. Cracking the password for test.zip

Here :

- -b = brute-force attack
- -c = Charset ‘a1’ (a for small alphabet and 1 for numeric value)
- -l = Length of password (min length-max length)
- -v = verbose (no compulsory)
- -u = Use unzip and path of password protected file

TABLE I : COMPARATIVE ANALYSIS OF THE ABOVE MENTIONED TOOLS

Sno.	Name of the Tool	Functionality/Working	Complexity
1	John the Ripper	• John The Ripper works by utilizing the dictionary strategy as that is one of the most effortless approach to break a password. • It takes text string tests from a word list utilizing regular dictionary words.. • John The Ripper tool cannot do Vulnerability Analysis . • JtR likewise limits repetition by just creation of guesses that vary from their quick archetype guess.	• Time taken by John The Ripper to crack a password primarily depends on the cracking mode(s) and the type of password protected file. • There are 3 modes namely-(i) Single crack • (ii)Wordlist mode • (iii)Incremental mode.
2	HashCat	• Hashcat uses precomputed word references, rainbow tables, and even a brute- force approach to manage find a practical and compelling way break passwords. • Hashcat has built-in benchmarking system, an integrated thermal watchdog, and support for over two hundred hash types, implemented with performance optimizations . • A lot of users prefer to use Hashcat Mask Attack instead of Hashcat-Brute Force Attack because mask attack tend to complete the process much faster and more efficiently by reducing the keyspace of passwords. • In Hashcat,Secret key portrayals are principally connected with hash keys, for example, MD5, SHA, WHIRLPOOL, RipeMD, and so forth. They are additionally characterized as a single direction work — this is a numerical activity that is anything but difficult to perform, yet hard to figure out.	• Hashcat offers numerous assault modes for acquiring compelling and complex inclusion over a hash's keyspaceA few of them are : (i)Dictionary Attacks (ii)Mask Attack (iii)Table-Lookup Attack. • A new version was released and one of the major updates was support for increased password lengths. The limit rose from a maximum of 15 characters to 55 (with some exceptions). • In hashcat-mask attack we can lessen the keyspace to 52*26*26*26*26*10*10*10*10*10 (237.627.520.000) blends. With a similar breaking pace of 100M/s, this requires only 40 minutes to finish.

3	Fcrackzip	• Fcrackzip which is an outsider tool for breaking compress documents passwords(zip files). It is the best apparatus as it attempts to look zipfile for scrambled records and attempts to figure their secret word. • Fcrackzip is an extremely effective apparatus and furthermore very simple to use for making a brute force assault on any compress document, for that diverse distinctive configuration for splitting the secret word of the compress record is utilized. So as to that , (- b) is used to brute force on that compress record, (- c) which characterize the charset for the word reference to brute force. • In fcrackzip, verbose is a mode which can be started utilizing (- v) boundary. Presently verbose mode creates expanded data. , verbose mode causes us to get data about the document in that secret key ensured compress record, similar to the size of that record, name of that document and so on.	• Longer passwords might seem to take longer to process and are thus more reliable. • The quick password cracker Fcrackzip reports that 204570 inspections per second on my device are made (measured under a single two W / O memory manager). • Naturally, it is more sluggish to write fcrackzip in C, and not in assembler. It's 12 percent slower when tested in a unix that is slightly loaded (same machine). • Default brute force begins on the specified starting password and successively attempts to print out all combinations, along with an approximately right indicator, until they are completed.

VIII. CONCLUSIONS

Password breaking is an undeniable danger. There are various strategies that can be utilized in an assault that have been depicted.[7] Approaches to forestall effective assaults by programmers for use by the two clients and executives have been examined. A usage of three password cracking tools were implemented to do comparison for effectiveness and their outcomes have been accounted for.All three tools are implemented on Linux platform and are available as open-source also.

The implementation of password cracking is done on both RAR/Zip files which were made for demo purpose. John The Ripper and Hashcat are pre-installed in Kali Linux Platform but Fcrackzip is an outsider tool. Password cracking has become a notable piece of entrance testing; white cap programmers will utilize password cracking instruments to attempt to break into secret data so as to test the quality.

Regardless of the way that passwords are scrambled before putting them away, the devices above can even now be utilized in breaking or uncovering the password. Despite the fact that these devices are generally successful against passwords that are simply scrambled and put away. These devices will be less successful against frameworks that utilize the strategies beneath to fortify the password.

IX. REFERENCES

1. Chester, J.A., 2015. Analysis of Password Cracking Methods & Applications.
2. Zviran, M. and Haga, W.J., 1999. Password security: an empirical study. Journal of Management Information Systems, 15(4), pp.161-185.
3. Feldmeier, D.C. and Karn, P.R., 1989, August. Unix password security-ten years later. In Conference on the Theory and Application of Cryptology (pp. 44-63). Springer, New York, NY.
4. Kumar, J. and Farik, M., 2001. Cracking Advanced Encryption Standard-A.
5. Kumar, S. and Agarwal, D., 2018. Hacking attacks, methods, techniques and their protection measures. International Journal of Advance Research in Computer Science and Management.
6. A Review of Top Open Source Password Cracking Tool, Federal University of Technology, Minna, Nigeria victor.yisa@futminna.edu.ng, babameshach01@futminna.edu.ng, olaniyi.emmanuel@st.futminna.edu.ng
7. Reasoning Analytically About Password-Cracking Software Enze Liu, Amanda Nakanishi, Maximilian Golla†, David Cash, Blase Ur University of Chicago, Ruhr University, alexliu0809, anakanishi, davidcash, blase}@uchicago.edu, maximilian.golla@rub.de
8. Spring 2015 Analysis of Password Cracking Methods & Applications John A. Chester The University Of Akron, jac177@zips.uakron.edu
9. John The Ripper :https://www.openwall.com/john/
10. Fcrackzip :https://github.com/hyc/fcrackzip
11. Hashcat, http://hashcat.net/oclhashcat/
12. Klein, D.V., 1990, August. Foiling the cracker: A survey of, and improvements to, password security. In Proceedings of the 2nd USENIX Security Workshop (pp. 5-14).
13. Hranický, R., Lištiak, F., Mikuš, D. and Ryšavý, O., 2019, July. On Practical Aspects of PCFG Password Cracking. In IFIP Annual Conference on Data and Applications Security and Privacy (pp. 43-60). Springer, Cham.
14. Liu, E., Nakanishi, A., Golla, M., Cash, D. and Ur, B., 2019, May. Reasoning Analytically About Password-Cracking Software. In 2019 IEEE Symposium on Security and Privacy (SP) (pp. 380-397). IEEE.
15. Aggarwal, S., Houshmand, S. and Flood, R., Florida State University Research Foundation Inc, 2016. Probabilistic password cracking system. U.S. Patent 9,438,419.
16. Valois, M., Lacharme, P. and Le Bars, J.M., 2019, June. Performance of password guessing enumerators under cracking conditions. In IFIP International Conference on ICT Systems Security and Privacy Protection (pp. 67-80). Springer, Cham.
17. Arends, R., Deussen, R., Green, B., Rush, J., Mache, J. and Weiss, R., 2018. Get a Clue: A Hands-On Exercise for Password Cracking. In Proceedings of the International Conference on Security and Management (SAM) (pp. 117-121). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).
18. Hranický, R., Zobal, L., Večeřa, V. and Matoušek, P., 2017. Distributed Password Cracking in a Hybrid Environment. In Proceedings of SPI (pp. 75-90).
19. Kakarla, T., Mairaj, A. and Javaid, A.Y., 2018, May. A Real-World Password Cracking Demonstration Using Open Source Tools for Instructional Use. In 2018 IEEE International Conference on Electro/Information Technology (EIT) (pp. 0387-0391). IEEE.
20. Liu, E., Nakanishi, A., Golla, M., Cash, D. and Ur, B., 2019, May. Reasoning Analytically About Password-Cracking Software. In 2019 IEEE Symposium on Security and Privacy (SP) (pp. 380-397). IEEE.