International Conference on Information and Communication Technology and Its Applications (ICTA 2016)
Federal University of Technology, Minna, Nigeria
November 28 – 30, 2016

A Review of Top Open Source Password Cracking Tools

Victor Legbo Yisa, Meshach Baba, and Emmanuel Tosin Olaniyi
Department of Cyber security Science, Federal University of Technology, Minna, Nigeria
victor.yisa@futminna.edu.ng, babameshach01@futminna.edu.ng, olaniyi.emmanuel@st.futminna.edu.ng

***Abstract*—Password authentication is one of the most common forms of authentication, verification and access control mechanism; Passwords remain the standard way to enhance the security of confidential information. Password cracking has become a salient part of penetration testing; white hat hackers will make use of password cracking tools to try to break into confidential information in order to test the strength. This write up review the most common forms of password cracking tools that can be used by penetration testers, the different methods of password cracking and ways passwords could be made more difficult to crack.**

Keywords-component; cracking; attack; open source; hashes

I. INTRODUCTION

Passwords, as a form of authentication can be said to be as old as time. In ancient times, watchmen would test those wishing to enter an area or approaching it to supply a watchword of which if correct entrance is given. In modern times however, a combination of username and password is a common means of authentication during log in processes.

User authentication, as defined by RFC 2828 is "the method of confirming an identity claimed by or for a system entity" [1].

Basically, the verification process is divided into four main categories: something known by user (knowledge factor) (e.g. password, PIN, answers to given questions), something the user owns (Ownership factor) (tokens, e.g. smart cards, electronic keycards, physical key), something that the individual is (Identity elements) (static biometrics, such as fingerprints, retina, face), and something the individual does (dynamic biometrics such as voice patterns, mouse movement pattern, handwriting, typing rhythm) [1] [2] [3]. Something the individual is and something the individual does can be categorized under inherent factors. The most common of the different authentication methods used now is the password authentication (something the individual knows) and has been commonly used as the line of defense against intruders [4] [2].

Password-based authentication works by comparing the credentials provided by a user with stored secrets. Because unauthorized users may have access to stored passwords, there is a need that passwords be encrypted during storage using cryptographic hash functions.

Password cracking can be defined as the recovery of plain password texts from a stored location which is usually encrypted. Password cracking is the process of obtaining the plaintext passwords from the stored encrypted secret, or at least an equivalent one [5]. Generally, password cracking in the hacking world ranges from decrypting password hashes stolen from a database to even hacking wireless networks.

This paper is organized as follows: section 2 will focus on the forms of password authentication while Section 3 will survey some password cracking techniques, Section 4 reviews the top password cracking tools and Section 5 shows the conclusion, followed by an appendix of abbreviations and available software.

II. FORMS OF PASSWORD AUTHENTICATION

There are various types of password-based authentication and each of them has their strengths and weaknesses when viewed in the area of memorability, usability and security. Despite the somewhat recurring idea in Computer Security that "the password is dead" and is therefore not recommended, it is still in use and will continue to be, at least for now. Some of these forms are discussed below:

A. Alphabetic Password

This is a type of password that entails the use of alphabetic characters **only** and could be either a dictionary word or not. An alphabetic password is very easy to remember by the users, which makes it relatively easy to be cracked either by a combination of social engineering and guessing or dictionary attack. A list of the most commonly used password types published by Google in 2013 [6], shows that this passwords were easy to guess and crack:

- Names within the house such as pet name, a close family members name or friend's name.
- Name of Birthplace or favorite holiday place.
- Names or things that is associated to their favorite sports club.
- The term "password" is also common.

B. Alphanumeric Password

This is a combination of alphabets and numbers in forming a password, it is the most common approach for authentication.

Although more secured than alphabetic passwords, alphanumeric passwords have its security and usability problems, one of which is the easy-to-guess substitutions such as 'A' for '4', 'S' for '5', 'E' for '3' and 'I' for '1', substitutions which attackers are conversant with. Another

drawback of the alphanumeric password is the difficulty of recalling the alphanumeric password by the user at the point of log in, especially if it is not frequently used or written down.

C. *Graphical Password*

Greg Blonder (1996) described graphical passwords [7] as involving the display of a predetermined graphical image and necessitating the user to select particular areas of the image in a particular order. The graphical password is a non-text based password, an alternative means of authentication intended for use in lieu of the conventional text-based passwords. Contrary to the memorability issues of alphanumeric passwords, graphical passwords are more memorable [8], and the relative ease or ability of humans to recognize faces and points within pictures has given it better usability when compared with other forms of password authentication, leading to imitations by machines with varying degree of success [9]. Graphical passwords, unlike the conventional alphanumeric password can be said to be even more secured as users do not have to write it down, making it less susceptible to social engineering attack.

In a 2010 study [8], Agarwal et al. compared alphanumeric passwords and graphical passwords in terms of memorability. They explained that in remembering passwords, password was inputted three times by each user in each trial, each user is only allowed to input correct password once; table 1 below shows the number of incorrect password submission.

In the analysis of memorability, MatLab was used and it can be seen that for the memorability factor (R1/R2/R3) and mode (alphanumeric/graphical), it was discovered that graphical password was always favored from the incorrect submission calculation. In a second experiment that compared the time for correct submission, a factor that can influence productivity, table 2 below shows that it took graphical password lesser time to summit correct when compared with alphanumeric password.

TABLE I. NUMBER OF INCORRECT SUBMISSIONS

	Mode	Mean R1 (SD)	Mean R2 (SD)	Mean R3 (SD)
No. of incorrect submissions	Alphanumeric	1.61 (1.63)	2.82 (3.93)	1.43 (2.76)
	Graphical	0.28 (0.82)	2.24 (2.77)	1.20 (2.52)

TABLE II. TIME FOR CORRECT SUBMISSIONS (IN SECONDS)

	Mode	Mean R1 (SD)	Mean R2 (SD)	Mean R3 (SD)
Time for correct submissions (seconds)	Alphanumeric	9.01 (4.56)	22.53 (13.32)	20.76 (17.58)
	Graphical	5.28 (1.70)	9.87 (3.91)	8.99 (3.43)

From table 1 and 2 above, it can be concluded that graphical password is more memorable when compared with alphanumeric password, which means it is generally more usable. However balancing usability and security seems to be almost impossible as researches on security and usability mostly support the notion that a system cannot be both usable and secured, but can only be one of them at a time [10].

Better memorability is the major advantage of graphical password over alphanumeric password, But, a major disadvantage is the fact that they are highly susceptible to shoulder-surfing [11].

A pivotal question that seems to be unanswered is: Is it possible to have a secured and usable authentication system? To be precise, does a secured, memorable and usable authentication technique for information security exist? Most probably the answer would be "maybe".

As it concerns the security, memorability and usability of password authentications, a few pointers are as follows:

- Avoid using any word as password from any dictionary.
- All good Passwords should contain special character, letters, and number and should not be less than eight characters long.
- Apply the pass-phrase approach in password generation i.e. for a phrase like "My much secured password is longer than 8 characters" the generated password would be "Mmspilt8c". This approach reduces the burden of memorability [12].

III. PASSWORD CRACKING TECHNIQUES

Password cracking tools can be majorly categorized into offline and online cracking categories.

Attacks such as dictionary and brute-force attack performed against on a live system login form or session is called online attack

The prevalence of online attacks may not be as much as offline attacks due to the fact that they are mostly impossible to pull off as there are numerous protection schemes in use that can make this kind of attack difficult and dangerous to realize but it is still possible to pull off if some of these mechanisms such as maximum unsuccessful authentication attempts and Captcha images can be evaded [13].

Offline attacks are carried after a password databases has been copied, or sniffed from an encrypted connection, offline do not alert the victim. This type of attack is popular as it is often easier to pull through as there are usually numerous possible vulnerabilities that can allow its exploit.

Dictionary Attack: Dictionary attack is a technique for exploiting a hashed authentication mechanism by trying to determine its decryption key by repeatedly trying thousands or millions of likely possibilities, such as words in a dictionary. In dictionary attack, wordlist comprising of possible and likely passwords is used by the cracker in attempting to gain access to a system [14] although wordlist that have proven to be the most successful in time past are composed of various public sources or databases filtered previously captured from real password [13]. There are several wordlists that are available, for Kali Linux users, there is a wordlist in the directory "usr/share/wordlists", with "rockyou.txt.gz" being the popular of all, and it can be unzipped and padded with more custom passwords or known weak passwords. While some are available for free, some are not, as they seem to contain even more language combinations than the free ones.

Brute-Force Attack: This type of attack, also known as exhaustive search involves the attacker trying every possible combination with the hope of eventually guessing correctly, it can be fast when used to check short passwords. Theoretically, a brute-force attack is a cryptanalytic attack that tries to decrypt all encoded data [15] (with the exception of the data encrypted in a secured-information theory).

The method involves computing the hash of the given password one by one, and then comparing the result of each hash with the target hash stored on the database. The drawback to this method is that the longer the password the longer the time to find the right password thereby consuming lots of system resources. Also, the computed hash cannot be reused to crack another password [16].

Hybrid Attack: This is a combination of both dictionary attack and brute-force attack, whereby the dictionary includes the wordlist and the brute-force is applied to each possible password in the list by taking each entry in the dictionary and creating a few variation of the dictionary word (like adding a prefix or suffix of numbers) [13] [17] . A Hybrid attack will also exponentially increase the computation and time depending on the amount of characters to be concatenated with the Dictionary entries [5].

Rainbow Tables: Hellman in 1980 introduced the time-memory-trade-off method used in reducing the time that is needed in cracking a cryptographic system [18] and was based on the fact that exhaustive search requires a lot of time or computing power to succeed.

Because of the drawback experienced in the Hellman time-memory-tradeoff Oechslin suggested what he termed rainbow tables (a time-memory-tradeoff technique) which drastically reduced the number of collisions experienced in the Hellman's model thereby reducing the number of calculations [19]. This is done by a pre-computation of the password hashes thereby reducing the time taking to crack a password. [2].

Rainbow Tables are faster than brute force attacks once the hash tables have been created, since the time it takes to compute the hash has been eliminated but it makes use of large storage area [9]. Another drawback of rainbow table is that it takes a lots of time to compute the rainbow table yourself.

IV. PASSWORD CRACKING TOOLS

A. *Ophcrack*

It is an open source program that decodes Windows logon password through rainbow tables by using LM hashes; it can also import the hashes from a variety of formats and sources even by directly dumping hashes from the SAM files of Windows. Most rainbow tables for LM Hashes are usually provided for free by the developers, although there are paid rainbow tables which tend to contain more hashes than the free counterpart. According to OPH Reviews, Ophcrack is fast and easy enough for a first-time password cracker user with basic Windows knowledge and it can crack most passwords within a few minutes, on most computers." Features

- Can be used on the most popular Operating systems including Windows, Linux/Unix and Mac OS X
- Can be used to breack LM and NTLM hashes
- Free tables available for Windows XP, Vista, 7, 8.1
- Brute-force module for simple passwords
- Audit mode and CSV Exports
- Analysis of passwords using real-time graphs.
- Live CD available
- Dumps and loads hashes from SAM encryption recovered from a Windows partition
- Its free and available for download

B. *Rainbow Crack*

Rainbow crack is a computer program that creates a rainbow tables for use in cracking the password; it works for general use by Philippe Oechslin faster time-memory trade-off technology [19]; and uses memory trade-off algorithm to crack hashes from the pre-computation of "rainbow tables". Well, it is time-consuming in pre-computing the tables but is considerably hundreds of time faster than a brute-force cracker once the pre-calculation is done. The only drawback noticed is that OS X is not supported.

- Time-memory tradeoff tool suites, including the production, sorting, conversion and lookup of rainbow tables.
- It is compatible with any rainbow table hash algorithm
- It is compatible with rainbow table of any character set, raw file format (.rt) and compact file format (.rtc)
- It supports computing on multi-core processor.
- GPU acceleration with NVidia GPUs and AMD GPUs (CUDA Technology)
- GPU acceleration on multiple CPUs
- Can run on both windows and linux operating systems
- Has both graphical interface as well as command line interface
- It has a merged rainbow table file format on all compatible operating systems

C. *Hashcat*

Hashcat is the self-proclaimed world's fastest CPU-based password recovery and cracking tool tool; although not as fast as its GPU counterpart oclHashcat, this seems to be the case as Fossbytes agrees. Hashcat can break 92672M h/s of hashes with the measurement made in hashes per second[20] There are available versions for popular operating systems: Linux, OS X and Windows and can come in either CPU-based or GPU-based variants. Its free for use and features [21].

- It is free for use
- Uses multiple GPU (up to 128gpus)
- Supports multiple Hash (up to 100 million hashes)
- Can be used on multiple Operating Systems
- Multi-Platform (OpenCL and CUDA support)
- Supports sessions, hex-salt, hex-charset, distributed cracking etc.
- Over 150 algorithms implemented with performance in mind
- Focused Dictionary based attacks
- Low resource utilization
- Built-in benchmarking system

- Integrated thermal watchdog and attack modes include [20]
- Straight and combination attack
- Brute-force
- Hybrid dictionary + mask
- Hybrid mask + dictionary

D. *Cain and Abel*

Cain & Abel is a password recovery tool for Windows OS that can be used by sniffing the network, deciphering encoded passwords using Dictionary, Brute-force and Cryptanalysis attacks, recording VoIP conversations, decoding scrambled passwords for the recycling of various types of passwords. It also helps recovering wireless network keys, revealing password boxes, cached password detection and analysis of routing protocols [22]. It is the ultimate MITM utility but is only available for Windows OS and can be a little complicated for novice users [23]. Its features include:

- Used for WEP cracking.
- Speeding up packet capture sped by wireless packet injection.
- Traceroute.
- ARP Spoofing
- Facility to record VoIP conversation
- Can be used to sniff Network Password
- It has IP to MAC addresses resolution facility
- It can crack diverse forms of hashes including but not limited to LM and NT hashes, IOS and PIX hashes, RADIUS hashes, RDP passwords, MD2, MD4, MD5, SHA-1, SHA-2, RIPEMD-160, Kerberos 5, MSSQL, MySQL, Oracle and SIP hashes.

E. *John the Ripper*

Originally developed by UNIX, John the ripper is a free software cracking tool to detect weak password and is now available for many flavors of UNIX, Windows, DOS and OpenVMS. It is one popular password testing and fracture program that combine a range of password crackers in a package that automatically detects types of password hash and can run against various encryptions [24] . While it is not designed specifically to crack strong passwords, it implements a brute-force strategy and brute-force as we know it, is considered infallible but can be time consuming and computationally expensive [25].

- Supports Dictionary and Brute-force attacks
- Multiplatform
- Its available for free

F. *THC Hydra*

The THC Hydra is a very fast and flexible network logon cracker which primarily employs a brute-force dictionary-based attack. Hydra supports a wide range of network protocols including but not limited to TELNET, FTP, HTTP, HTTPS, SNMP, IMAP, POP3, etc. It provides a Command Line Interface and a Graphical User Interface. Its features include

- Available for Windows, Linux and OS X
- It is extensible and easy to add new modules
- Very effective against remote authentication services.
- Can perform rapid dictionary attacks against more than 30 protocols.
- Supportive with Brute-force and Dictionary attacks

G. *L0phtCrack*

Is a different flavour of OphCrack that tries to crack Windows password from hashes by using Windows workstations, network servers, primary domain controllers and Active Directory for cracking passwords; using dictionary and brute-force attack to generate and guess passwords.

Lophtcrack has the following features and abilities

- Extraction of hashes from Windows versions, multiprocessor algorithms, and networks monitoring and decoding.
- It runs On most BSD and Linux variants with an SSH daemon.
- Can run on windows XP and higher operating system, runs on windows server 2003 and 2008 and in both 32 and 64 bit environments
- Can remotely retrieve passwords
- Can perform scheduled scans
- Scoring of passwords
- Supports pre-calculated dictionary wordlist
- Supports Unix & Windows password
- Executive Level Reporting
- Can give information on the risk status of Passwords
- Password Audit Method

V. PROTECTING AGAINST PASSWORD CRACKING

Despite the fact that passwords are encrypted before storing them, the tools above can still be used in cracking or revealing the password. Although this tools are mostly effective against password that are just encrypted and stored. These tools will be less effective against systems that employ the techniques below to strengthen the password.

A. *Salting*

This involves adding some bits of information known as salt to a password before they are hashed [26], making it un-guessable or more difficult for a standard rainbow table to crack [27]. When two salts are used, it becomes harder to crack the password [28]. The use of salts will prevent the use of rainbow tables in order to break password hashes. Although it is easy to implement and straight forward, it is also important to salt passwords in a proper and orderly manner. For example for every password or user, a different salt should be created so that a rainbow table will not be created for the set of passwords. Also a large salt value will be more preferable to smaller ones and salt values should be randomly generated [29].

B. *Strong password*

The use of passwords that contains both capital and small letters, numbers and special characters and a total of at least 8 characters or more can affect the effectiveness or greatly increase the time it takes to crack this types of password.

C. *Hybridized Authentication*

Password form of authentication can be combined with other forms of authentication such as biometric, tokens or cards, thereby making these cracking tools less effective in password cracking

VI. CONCLUSION

Password authentication remains the mostly used method of verification; however there are several vulnerabilities in its use (such as password reuse, dictionary words etc.) and several classes of attacks against passwords. These vulnerabilities can easily be exploited by password cracking tools. Most of these password cracking tools are available for free or in open source licenses. Based on the cracking task, a penetration tester may adopt any of these cracking tools as suitable (based on the tools features and characteristics) to carry out his pen testing task. An ethical hacker and penetration tester can pick from any of this open source cracking tools for pure authorized cracking purposes and it is strongly advice that these tools be used for learning purposes.

REFERENCES

[1] R. Shirey, "RFC 2828: Internet Security Glossary," The internet Society 13, 2000.

[2] A. L.F Han, D. F. Wong, and L. Chao, "Password Cracking and Countermeasures in Computer Security: A Survey," arXiv preprint arXiv:1411.7803., November 2014.

[3] D. Dasgupta and S. Saha, "A biologically inspired password authentication system," in CSIIRW '09 Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies, New york, 2009, p. 41

[4] D. Seeley, "Password cracking: A game of wits," in Communications of the ACM 32.6, 1989, pp. 700-703.

[5] S. Marechal, "Advances in password cracking.," Journal in computer virology, vol. 4, no. 1, 2008 pp. 73-81.

[6] Techlicious / Fox Van Allen @techlicious (2013-08-08). "Google Reveals the 10 Worst Password Ideas | TIME.com". Techland.time.com. Retrieved 2016-10-19

[7] G.E.Blonder, . U.S. Patent No. 5,559,951. Washington, DC: U.S. Patent and Trademark Office, 1996.

[8] G. Agarwal,.and R.S. Shukla,. Security Analysis of Graphical Passwords over the Alphanumeric Passwords. Int. J. Pure Appl. Sci. Technol. 1(2), 2010,pp 60-66

[9] R. Chellappa, C. L. Wilson, and S. Sirohey, "Human and Machine Recognition of Faces: A Survey," Proceedings of the IEEE, vol. 83,1995, pp. 705-741

[10] L. F. Cranor and S. Garfinkel, "Secure or Usable?," IEEE Privacy & Security, vol. 2, 2004, pp. 16-18

[11] F. Tari,, A. Ozok, and S.H. Holden, . A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. In Proceedings of the second symposium on Usable privacy and security 2006, pp. 56-66. ACM.

[12] J. Yan, A. Blackwell, R. Anderson, and A. Grant, "Password Memorability and Security: Empirical Results," IEEE Privacy & Security, vol. 2, 2004, pp. 25-31

[13] C. Yiannis, "Modern Password Cracking: A hands-on approach to creating an optimised and versatile attack.," Surrey, Thesis 2013.

[14] Y.S. Dandass, , "Using FPGAs to parallelize dictionary attacks for password cracking," in Hawaii International Conference on System Sciences, Proceedings of the 41st Annual, Hawai, 2008, pp. 485-485.

[15] C. Paar and P. Jan, Understanding cryptography: a textbook for students and practitioners.: Springer Science & Business Media, 2009.

[16] H. Kumar et al., "Rainbow table to crack password using MD5 hashing algorithm," in IEEE Conference Publishing School of Information Technology and Engineering (SITE) VIT University, Vellore, India, JeJu Island, 2013, pp. 433-439.

[17] nFront security. (2011) Everything Administrators need to know about Windows password security. [Online] http://nfrontsecurity.com/downloads/nFront-Security-Whitepaper-Everything-You-Need-To-Know-About Password-Security.pdf

[18] M. Hellman, "A cryptanalytic time-memory trade-off," IEEE Transactions on Information Theory, vol. 26, no. 4, pp. 401-406, 1980.

[19] P. Oechslin, "Making a faster cryptanalytic time-memory trade-off," in Annual International Cryptology Conference., Berlin, 2003, pp. 617-630.

[20] J. A. Chester,., "Analysis of Password Cracking Methods & Applications" (2015). Honors Research Projects. Paper 7.

[21] Hashcat. Advanced Password recovery. [Online]. https://hashcat.net/hashcat/

[22] A. E. .Mohamed. (2013, January) Password Cracking Using Cain & Abel. [Online]. http://resources.infosecinstitute.com/password-cracking-using-cain-abel/

[23] S. Fahmy, N. Akhyari , and N. Shamsuddin, "Wireless network attack: Raising the awareness of Kampung WiFi residents," in Computer & Information Science (ICCIS), 2012 International Conference on, vol. 2, 2012, pp. 736-740.

[24] S Balasubramanian. (2010, July) Techulator. [Online]. http://www.techulator.com/reviews/133-John-The Ripper.aspx

[25] R. Lim, "Parallelization of John the Ripper (JtR) using MPI," University of Nebraska, Nebraska, 2004.

[26] M. Abadi, T. Mark A. Lomas, and R. Needham, "Strengthening passwords," d i g i t a l Systems Research Center, Palo Alto, California, Technical Note September 1997.

[27] B.Groza, "Analysis of a Password Strengthening Technique and Its Practical Use," in 2009 Third International Conference on Emerging Security Information, Systems and Technologies, Athens, Glyfada, 2009, pp. 292-297.

[28] U. Mamber, "A simple scheme to make passwords based on one-way functions much harder to crack. Computers & Security, 15(2), 171-176.," Computers & Security, vol. 15, no. 2, pp. 171-176, 1996.

[29] K. Brown, "The Dangers of Weak Hashes," SANS Institute Infosec reading room, GIAC GWEB Gold Certification November 2013.