PASSWORD GENERATION STRATEGY

Predrag Tasevski

Tartu University, Faculty of Mathematics and Computer Sciences, major: Master of Science in Cyber Security

Abstract. Nowadays attacking the passwords is one of the most straightforward attack vectors, which authorize access to information system. Numerous methods are feasible to perform, attempt to guess or crack passwords, with a different methods, approaches and tools. This paper analyses the possibilities of using the tools and example to accomplish the password guesses in many unlike methods with pronounced noteable tests and statistic results. The overall service to the follower is to insure for the potential needs: preventing password cracking, information security audit, password recovery, security policy and etc.

INTRODUCTION

Access control to information systems is often implemented via passwords; hence attacking the passwords is one of the most straightforward attack vectors. Typical computer users nowadays may require passwords for many purposes: logging in to the system accounts, retrieving e-mail from servers, accessing programs, databases, networks, web sites, e-banking and etc. Furthermore, a password is a secret word or string of characters that is used for authentication, to prove identity or gain access to a resource. The password should be kept secret from those not allowed access. Password or watchwords have been used since ancient times in to the Roman military system. [5]

Password cracking is a process of attempting to guess or crack password to gain access to a system. In addition, can be a process of recovering passwords from data that has been stored in the system. In summary, password cracking common approach is to repeatedly try to guess the password. Moreover, the purpose of password cracking might be to help a user recover a forgotten password, gain unauthorized access to a system, or preventive measure by system administrator to check for passwords that are easy to crack.

As a consequences, this research investigates the usage of password cracking tools, methods and approaches that can be used in guessing the passwords, examples of leaks and generating password dictionaries, statistic of already cracked passwords from available password dictionaries and test.

In addition, this research gives an approach of performing a password cracking techniques not only to on-line and offline, but to the file system on real time on-the-fly encryption software application TrueCrypt[1], which creates a virtual

[1] TrueCrypt: http://www.truecrypt.org/

encrypted disk within a file or a device-hosted encrypted volume on either an individual partition or an entire storage device. [14]

Therefore, the tests that have been done during the paper were as expected. Moreover, the tests were done with a system user hashes passwords and virtual encrypted disks. The both guesses were made with the leaked input dictionaries with a different methods and tools. The paper concludes with the results of a tests performed during the passwords guesses, where the first is with simple password and the other test with strong password.

1 METHODS

Password cracking is a method of guessing attack. An attacker makes guesses about the user's passwords until they guess correctly or they give up.

In this manner, methods of passwords cracking can be paraphrase as a test of passwords cracking, because we do not know if the proper method/test is going to be efficient. Hence, we are going to see the different methods of performing cracking passwords.

There are three basic types of password cracking methods that can be automated with tools [15]:

- Dictionary - A file of words is run against user accounts, and if the password is a simple word, it can be found pretty quickly.
- Hybrid - A common method utilized by users to change passwords is to add a number or symbol to the end. A hybrid attack works like a dictionary attack, but adds simple numbers or symbols to the password attempt.
- Brute force - The most time-consuming, but comprehensive way to crack a password. Every combination of character is tried until the password is broken.

In other words, dictionary and the hybrid methods are ad-hoc models, which indicate methods by the use of dictionary. Before an attacker could rely on simple brute force methods and ad-hoc models, there is a growing demand for more effective ways to predict the user's password; there are tools and techniques that can be used like a rainbow tables, dictionary based attacks and probabilistic password cracking.

The case of dictionary attacks is when the attackers use a dictionary comprised of words, were they suspect that the target may have been used in their password. The attacker then applies colander rules to these input words, such as capitalization the first letter, adding three digits to the end, changing the letter 'a' to '@' and etc. to further match the targets password. The important thing is to remember that the success of a dictionary base attack depends not only on the input dictionary selected, but also on the word colander rules applied. The attackers first try with the small input dictionary, if this fails attacker crack the password using a much larger input dictionary. In addition to the second section Examples and Tools there will be a discuss about the tools and the techniques that can be performed in oreder to create a good base input dictionary.

Brute force attacks are very common in most password cracking tools. It is a method that does not use any input dictionary of human generated words. These attacks are popular because the most input dictionaries only cover a fraction of the total words that are made by users creating their passwords.

There are four types of attacks that can be performed during the brute force attacks methods:

- Pure brute force is brute force attack that does not use outside probability information that is not found inherently in the key-space being searched. [6]
- Letter frequency analysis attack is an attempt to use the frequency of characters appearing in a training set to increase the effectiveness of a brute force attacks. [13]
- Markov models in password cracking is a way to represent the joint probability of different characters appearing together. [3]
- Targeted brute force attacks can comprise letter frequency analysis and Markov models, but applies outside logic to these attacks. [16]For example performing letter frequency analysis attacks, but use a different character set for each character position.

Rainbow tables reduce the difficulty in brute force cracking a single password by creating a large pre-generated data set of hashes from nearly every possible password. Rainbow tables and RainbowCrack are the result of the work and subsequent paper by Philippe Oechslin [12]. The main benefit of rainbow tables is that while the actual creation of the rainbow tables takes much more time than cracking a single hash, after they are generated you can use the tables over and over again. Additionally, once you have generated the rainbow tables, RainbowCrack is faster than brute force attacks and needs less memory than full dictionary attacks. [4]

Moreover rainbow tables are based on the idea of hash chains where the important concept is the index value. In a standard offline password cracking attack, the attacker possesses a password hash, and is attempting to guess the password that created it. That is why rainbow tables can be best thought of as a very efficient, but lousy, compression algorithm for hash lookup tables. The index value ranges from 0 to (key max-1). This is an example, if the attacker was trying to brute force all words seven character long which contains only lower cases letters the key max would be 26^7. There are three main functions in creating and application of rainbow tables: *IndexToPlain*, *PlainToHash*, and *HashToIndex*. [8]

2 EXAMPLES AND TOOLS

Although there are many existing tools available for password cracking, the difference between these tools is not the technique they employ but the password types they support. Second consideration is a distribution of the speed, which tools can make guesses and the hardware tools that are taken into the account. Many password cracker tools can perform their hash calculation on

CPU (Core Processor Unit), GPU (Graphical Processor Unit), or FPGAs (Field Programmable Gate Arrays). Most of those tools can perform on-line or offline cracking passwords. This paper will examine the proposed two types of methods, with the examples and tools that can be employed, the most common tools.

Programs such as THC Hydra[2] , and NCrack[3], are specifically tailored to attack network services and on-line websites. These programs are optimized to perform on-line password cracking attacks with network scanning ability and other features built into. Because they perform on-line attacks, these tools were also generally run by the use of very small input dictionaries due to the fact that they were often only allowed a few guesses against each on-line target. [17]

For instance tools that can perform an offline password cracking attack are listed below with arguments and notes:

- John the Ripper – it is the one of the oldest but still maintained password cracking program. It uses Unix Based Crypt hashes. It is an open source project, the advantage is that supports a pipe guesses, which means it is possible to write a custom algorithm to generate password guesses and then use it as a backend cracker. Also it includes the ability to export guesses generated from the built in algorithms to other programs, which made it convenient to map the effectiveness of a password cracking session by keeping track of exactly how many guesses were required to crack each password. [6]
- Cain & Able – it runs on windows operation systems, is free, it is graphical user friendly. One of the popular future is that can be used as network sniffer that automatically grabs passwords and password hashes that it sees. Cain & Able is not only a password cracking program, but it is also highly effective at collecting passwords and password hashes from targets on the local network. It has been built as support for creating Rainbow Tables, and has the ability to submit password hashes to on-line hash lookup databases. It supports a brute force methods only support letter frequency analysis attacks which is very limiting. The Cain & Able is a very well designed program; it is only useful for cracking weak passwords. [11]
- L0phtcrack – in 2000 year it was one of the first password cracking programs that could attack Windows LM hashes (LAN Manager Hash). It is most used for professional penetration testers who are performing security audits on company networks. Therefore it has a very well designed GUI[4], and the ability to create executive reports. L0phtcrack puts its emphasis on performing standardized attacks as part of a risk assessment. It is not designed to crack strong passwords. [9]
- Elcomsoft Distributed Password Recovery - was one of the first companies to produce a password cracking program that not only could be distributed across multiple computers, but also takes advantage of a computer's GPU, to hash password guesses as well. Can be used to crack the windows log-in passwords and encrypted files. Does not allow specifying a

[2] THC-Hydra - `http://www.thc.org/thc-hydra/`
[3] Ncrack - `http://nmap.org/ncrack/`
[4] GUI - Graphical User Interface

custom word rules for a dictionary based attacks. For brute force attacks, it only supports letter frequency analysis and not the more advanced techniques such as Markov modeling or targeted brute force attacks. [10]

- AccessData Password Recovery Toolkit – can be used for cracking file encryption and password hashes. It is designed to work with field programmable gate arrays (FPGAs), instead of GPU to speed up password cracking attack. It is a fully customizable method for creating dictionary and brute force style attacks. For the brute force attacks, it supports both Markov modeling and targeted brute force; it allows brute force and dictionary attacks to be interlaced, where it switches between the two depending on the attacker's methodology. Only downside is that does not allow guesses generated by outside programs to be used as input like John the Ripper does. [1]
- TC Brute - it is a TrueCrypt bruteforcer. It depends on a word list and works with multi threaded crack action. [7]The tool is powerful and it comes with a GUI which runs only on Windows platforms. Due to the testing period we came with an conclusion that the word list has to be good constructed to be able to guess the passwords.

Finding and creating input dictionaries is a common peocess. There are many dictionaries available that can be downloaded from Internet or tools that can be used to create input dictionaries, dictionaries which are specifically created for password cracking attacks, and with user names. One of the main lists that we have found, the last modification was made on 3rd January 2011. The web site is http://www.skullsecurity.org; where Skull Security is a website written by Ron Bowes. In the site you can find dictionaries that come with tools/worms/etc., designed for cracking passwords and input dictionaries passwords that were leaked or stolen from sites; Miscellaneous non-hacking dictionaries can be found, which are dictionaries of words but not of passwords, they may be a useful for one reason or another and Facebook list passwords based on the directory available from https://www.facebook.com/directory/.

There are other tools that can be used to generate input dictionaries from Wikipedia, or other sources. The first try was in a WikiGrabber command line tool that builds custom dictionaries by spidering/crawling [19] web pages hosted on Wikipedia or other WWW[5]. Creating custom dictionaries based on Wikipedia articles actually turned out to be a very difficult problem though as it was hard to construct the appropriate search queries. Other source can be used a sister project of Wikipedia, Wiktionary[6], and is used to generated dictionaries for different languages. The main disadvantage to this approach of creating dictionaries is that requires a big amount of space of hard drive. There is another example of Python source code developed for password dictionary generator by Travis Altman. Where Altman is giving a perfect example of how much time and space it does require to create a dictionaries with a different range of characters, length and line size. [2]

[5] WWW - World Wide Web

[6] Wiktionary - `http://www.wiktionary.org/`

Furthermore, the DRCrack[7] is an application dedicated to dictionary based rainbow table password cracker, (known as drcrack). The original source code is based off of rcrack[8] written by Zhu Shuanglei. Drcrack allows the creation and use of dictionary based rainbow tables. For example, you could create a rainbow table that would attempt to crack all passwords of length one through six, containing alphanumeric characters. There is a good description and documentation how to perform the tasks on the web site of the project. Or there is an on-line creating rainbow table calculator site, created by Steve Thomas in 2010: http://www.tobtu.com/rtcalc.php. Analogous, an Objectif Sécurité[9] has developed an open source applications using rainbow tables, for cracking the office documents or system passwords.

Unlike the dictionary and the brute force attack, probabilistic password cracking assumes that not all possible guesses have the same probability. If passwords can be guessed in a decreasing order of probability, this would lead to passwords being cracked with a lower number of guesses which therefore increases the efficiency of the password cracking process. The probabilities of passwords are calculated systematically from an existing list of plain-text passwords which measures the frequencies of certain patterns and the characters that are used. [18]

3 STATISTICS

the previous information explains which methods and what tools can we use to execute a password cracking attacks. What input dictionaries we should use and how to perform a rainbow tables attacks with a practical tools. In the above example, web site from Ron Bowes; skullsecurity.org we can find merit statistic, tests that were made by the author from a various dictionaries against the different sets of leaked passwords.

Specifically the test and the charts were based on the most common input dictionaries that were used for performing a password cracking attacks. Totally eight charts, different input dictionaries comparing with the others in a combinations of the amount of the list size over the number of successful cracked passwords.

First with the top worst passwords list available to download, compared with the other bigger size dictionaries, Figure 1; Shows that the phpbb list dictionary with almost the same size list it guesses most passwords. Almost around 450 number of cracked.

Chart Elitehackers and Hak5 are password generated lists from a part of a zf05.txt file. Because it was with a big size dictionary, that some of the tools have limitation for input dictionaries that they can execute password

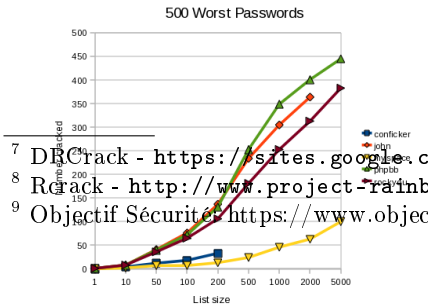

Fig. 1. 500 Worst Passwords

[7] DRCrack - https://sites.google.com/site/reusablesec2/drcrack
[8] Rcrack - http://www.project-rainbowcrack.com/
[9] Objectif Sécurité https://www.objectif-securite.ch/en/index.php

cracking attacks. That is why were divided with a two groups, elitehackers with a less size and hak5 with a bigger list size. Figure 2

Next chart is with a religious password list dictionary. It is leaked out with a file name of faithwriters.txt. Comparing with another list it shows that it cracked around 1000 passwords. In Figure 3 we can see that this time the rockyou list conquest the phpbb list with a few more numbers of cracked passwords. On the previous figure shows how the phpbb list was always few numbers better in a number of cracked passwords then the other list dictionaries.

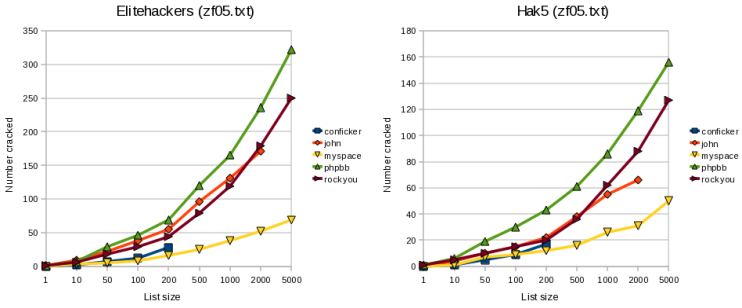

Fig. 2. zf05.txt

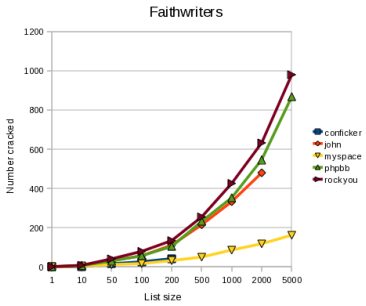

Fig. 3. Faithwriters

Below, Figure 4 which compares the most known sites, leaked passwords list from well-known on-line services that most of us are using every day. Varieties shows how different are four dictionary password lists by their content, how successful they have performed a number of cracked passwords. In the most cases the par excellence is a phpbb list, compared to the other input dictionaries lists. Only in the Figure 4, chart 4, the Hotmail list password dictionary is winning over the rockyou list. Comparing the sizes of these dictionaries is between the 250MB to less than 90KB.

Above all, statistics and the charts were not made from all dictionaries that were leaked on the Internet. As a result, only the main ones were mentioned, the

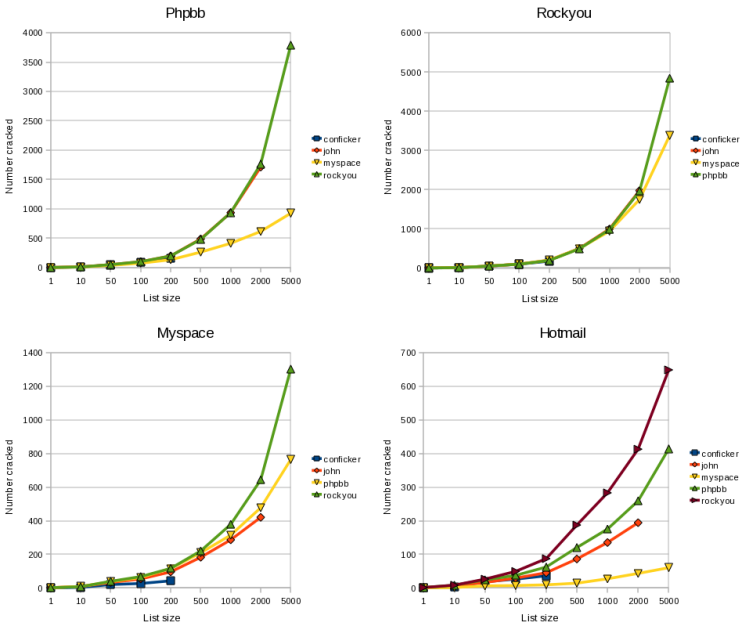

Fig. 4. Popular dictionaries

first leaks, and used to perform a numerous password cracking attacks, with a successful password guesses.

4 TEST

Complementing those study statistics, next step are tests that were performed to make the conclusion of the paper. Setup of the instance were performed by the John the Ripper, Cain & Able and TC Brute tools. In order to get a broader idea of performing password cracking, the tests were separated in two approaches.

First approach is with the tools John the Ripper and Cain & Able. Where the system hash password file were tested of performing password cracking with an input dictionaries and different character password strength with different methods.

Second approach is with the tool TC Brute where two virtual drives were encrypted with different passwords by the TrueCrypt application. Brute force method performed with different input password dictionaries and different password strength.

Analogous to the both tests and different methods, input dictionary and brute force password cracking were pull through successfully. As a result it still depends on circumference of passwords strength that were used for conducting this tests and the length of input dictionaries, in other words, obfuscation between them.

CONCLUSION

People notoriously remiss at achieving sufficient entropy to produce satisfactory passwords, thus attacking the passwords is one of the most straightforward attack vectors. Password cracking has been around ever since someone invented first secret word. On the contrary, many odds of methods and the techniques that can conduct to password cracking, in on-line and offline environment; tools that can accomplish to guess the passwords for differential goals; dictionaries that are leaked from on-line services and generated by others and tools that can help with generating a password input dictionaries with a different languages. It infer with the measures that should be inlaid to a better password strength, policy and protection. As a practical matter, passwords must be both reasonable and functional for the end user as well as strong enough for the intended purpose.

By way of illustration, this paper introduced to the usage of password cracking tools, methods and approaches that can be perform in guessing the passwords, examples of leaks and generating password dictionaries, statistic of already cracked passwords from available password dictionaries and as a consequence of test results.

The summary, is to acquaint the follower about the eventuality methods, tools and input password dictionaries that are available, tests that were with an positive cracking password results. To insure for the potential needs: preventing password cracking, information security audit, password recovery, security policy and etc.

ACKNOWLEDGMENT

This research paper would not have been possible without the support and encouragement of my colleagues and friends. Great thanks to Tartu University and the Tallinn Technical University who enroll me with a full scholarship in the master of cyber security studies and DoRa9 scholarship funded by Archimedes Foundation.

References

1. AccessData. Password recovery toolkit® (prtk®). AccessData Decryption Tools, 2011.
2. Travis Altman. Password dictionary generator. http://travisaltman.com/, 2010.
3. Shmatikov Arvind, Narayanan; Vitaly. Fast dictionary attacks on passwords using timespace. Technical report, The University of Texas at Austin, 2005.
4. Chris Gates. Tutorial: Rainbow tables and rainbowcrack. Tutorial, 2011.
5. N.S. Gill. The roman military system. Web article, 1997.
6. The OpenWall Group. John the ripper password cracker, 2010. Openwall Project - Information Security software for open environments.
7. IsNull. Tcbrute, July 2010.
8. Kestas Chris Kuliukas. How rainbow tables work. kestas.kuliukas.com; Kestas home page, 2006.

9. LLC L0pht Holdings. L0phtcrack password auditor. L0phtCrack Password Auditor, 2009.
10. ElcomSoft Co. Ltd. Elcomsoft products, 2011.
11. Massimiliano Montoro. oxid.it web site. oxid.it web site, 2011.
12. Philippe Oechslin. Making a faster cryptanalytic time-memory trade-off, 2003.
13. L. Stitson. Intro to cryptography notes, 7 2003.
14. Coorporation: TrueCrypt. Truecrypt - free open-source disk encryption - documentation, March 2011.
15. Russell Dean Vines. Ethical hacking tools and techniques: Password cracking. searchsecuritychannel.techtarget.com, 2007.
16. CHARLES MATTHEW WEIR. Middlechild password cracker. Reusable Security Tools, 2010.
17. CHARLES MATTHEW WEIR. *Using Probabilistic Techniques To Aid In Password Cracking Attacks*. PhD thesis, The Florida State University, 2010.
18. Matt; Sudhir Aggarwal; Breno de Medeiros; Bill Glodek Weir. Password cracking using probabilistic context-free grammars. Technical report, Internet Security Seminar, 2010.
19. Wikipedia.org. Web crawler. Wikimedia.org, 2011.