PASSWORD ATTACKS AND GENERATION STRATEGIES

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. There are numerous feasible methods, attempt to guess or crack passwords, with a different methods, approaches and tools. This paper analyzes the possibilities of using the tools and gives an example of how to accomplish the password guesses in different methods with tests which can be demonstrated together with comparison of input dictionary lists. The overall service to the follower is to insure for the potential needs: preventing password cracking, information security audit, password recovery, security policy 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 require passwords for purposes: logging into the system accounts, retrieving e-mail from servers, accessing programs, databases, networks, web sites, e-banking etc. Furthermore, a password is a secret word or string of characters that is used for authentication in order to prove identity or gain access to a resource. The password should be kept secret from those which do not have allowed access. Password or watchwords have been used since ancient times in the Roman military system [6].

Password cracking is a process of attempting to guess or crack password to gain access to a system. It can also be a process for recovery of passwords from data that are stored into the system. Therefore, password cracking is an approach which uses repetition in order to try and guess the password. Moreover, the purpose of password cracking might be to help a user to recover a forgotten password, gain unauthorized access to a system, or perform preventive measure by system administrator to check for passwords strength.

In fact, this paper examines the usage of password cracking tools, methods and approaches that can be used in guessing the passwords, examples of leaks and generating password dictionaries, comparison 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 encrypted disk within a file or a device-hosted encrypted volume on either an individual partition or an entire storage device [17].

Therefore, the tests were carried out during the writing of this paper ended with expected results. Indeed, the tests were examined with a system user hash passwords and virtual encrypted disks, through the input of leaked dictionaries with a different methods and tools. The paper concludes with the results of a tests which were performed during the passwords guesses. The first is with simple password and the other with strong password.

1 METHODS

Password cracking is a method of guessing the 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 for passwords guessing, 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 automatize with tools [18]:

- 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 are methods by the use of dictionary. Before an attacker could rely on simple brute force methods and ad-hoc models, there are growing demands for more effective ways to predict the user's password, e.g. 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 '@' etc. to further match the targets password. The important thing 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. Attacker first tries with little input dictionaries, if this fails, then the attacker crack the password using a much larger input dictionaries.

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

Brute force attacks are 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 [7].
- 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 [16].
- Markov models in password cracking is a way to represent the join probability of different characters appearing together [15].
- Targeted brute force attacks can comprise letter frequency analysis and Markov models, but applies outside logic to these attacks [19]. For example performing letter frequency analysis attacks, but use a different character set for each character position.

Rainbow tables use a refined algorithm by using a number of different reduction functions to create multiple parallel chains within a single "rainbow" table, reducing the probability of false positives from accidental chain collisions, and thus increasing the probability of a correct password crack. It also increases the probability of a correct crack for a given table size, the use of multiple reduction functions also greatly increases the speed of look-ups [9].

Rainbow tables are specific to the hash function they were created for e.g., MD5 tables can crack only MD5 hashes. The theory of this technique was first pioneered by Philippe Oechslin as a fast form of time-memory trade-off [14], which he implemented in the Windows password cracker Ophcrack. The more powerful RainbowCrack program was later developed that can generate and use rainbow tables for a variety of character sets and hashing algorithms, including LM hash, MD5, SHA1, and NTLM [9].

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 [14]. 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 [5].

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 thought of as a very efficient, compression algorithm for hash lookup tables. The index value ranges from 0 to (key max-1). For example, if the attacker was trying to brute force

all seven character long words 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* [10].

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 as well as 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 passwords cracking. 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 are 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 [20].

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 programs. It uses Unix Based Crypt hashes. It is an open source project, its advantage is that it 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 has 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 exact number of guesses which are required to crack each password [7].
- Cain & Able – it runs on windows operation systems, it is free and its graphic interface is user friendly. It 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 with letter frequency analysis attacks which is very limiting [13].
- L0phtcrack – in 2000 year it was one of the first password cracking programs that could attack Windows LM hashes (LAN Manager Hash). It is

[2] THC-Hydra - `http://www.thc.org/thc-hydra/`
[3] Ncrack - `http://nmap.org/ncrack/`

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 [11].

- Elcomsoft Distributed Password Recovery - was one of the first companies to produce a password cracking program. It can be distributed across multiple computers, and takes advantage of a computer's GPU to hash password guesses. It is used to crack the windows log-in passwords and encrypted files. It does not allow specifying a 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 [12].
- 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 method which is fully customized 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, where it switches between the two depending on the attacker's methodology. Its 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 [8]. The tool is powerful and it comes with a GUI which runs only on Windows platforms. Due to the testing period we came up with a conclusion that the word list should be well constructed in order to be able to guess the passwords.

Finding and creating input dictionaries is a common process. There are many dictionaries available that can be downloaded from Internet or, on the other hand, tools that can be used to create input dictionaries. These dictionaries are specifically created for password cracking attacks. One of the main lists that we have found, is Skull Security [3] a website written by Ron Bowes. In the web page you can find dictionaries that come with tools/worms/etc., designed for cracking passwords and input dictionaries passwords that were leaked or have been stolen from sites; Miscellaneous non-hacking dictionaries can be found, which are dictionaries of words but not of passwords, they may be useful for one reason or another. Also there are Facebook list passwords based on the directory available from service Facebook User Directory [4].

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 [22] 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 because it was hard

[4] GUI - Graphical User Interface
[5] WWW - World Wide Web

to construct the appropriate search queries. Other source that can be used is a sister project of Wikipedia, Wiktionary[6], and is used to generate dictionaries for different languages. The main disadvantage to this approach of creating dictionaries is that it 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].

Furthermore, the DRCrack[7] is an application dedicated to dictionary based on rainbow table password cracker, (known as drcrack). The original source code is based on 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 from 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. Analogously, 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 [21].

3 COMPARISON OF INPUT DICTIONARY LIST

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 [3], 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: 500 worst passwords, hak5 (zf05.txt), elitehackers (zf05.txt), faithwriters, phpbb, rockyou, myspace and hotmail list. The main aim is to show the difference and content of available input dictionary lists and the ways in which successful attempts were made during the estimations of guessing a passwords.

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

First, to compare the top worst list of passwords with one of the best list, dictionary, where seen in Figure 1 that phpbb list with almost the same size of a list of passwords is attacked with about 450 pieces of cracked passwords, compared with the other lists from chart.

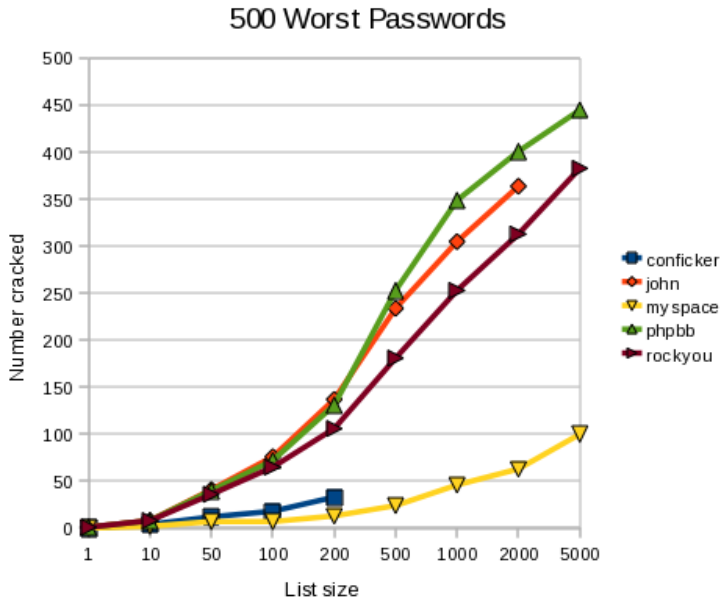

Fig. 1. 500 Worst Passwords list

Chart Elitehackers and Hak5 are password generated lists from a part of a zf05.txt file. Because it is with a big size dictionary, some of the tools have limitations for input dictionaries to execute password cracking attacks. That why it is divided into two groups, elitehackers (Figure 2) with a less size and hak5 (Figure 3) with a bigger list size.

Next chart is with a religious password list dictionary. It is leaked out with a file name of faithwriters.txt. If we compare it with another list it shows that it cracked around 1000 passwords. In Figure 4 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.

Figures 5, 6, 7 and 8 present a comparison of most of the well known sites/online services. Varieties shows the amount of difference in these four dictionary password lists in respect to their content, how successful they are in the number of cracked passwords. In the most cases the par excellence is a phpbb list, Figure5, compared to the other input dictionaries lists. Only in the Figure 8 hot-

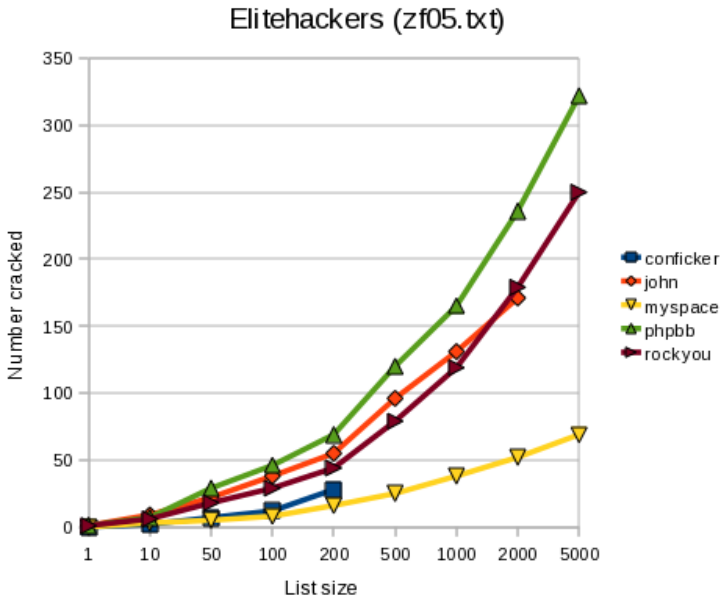

Fig. 2. Elitehackers (zf05.txt)

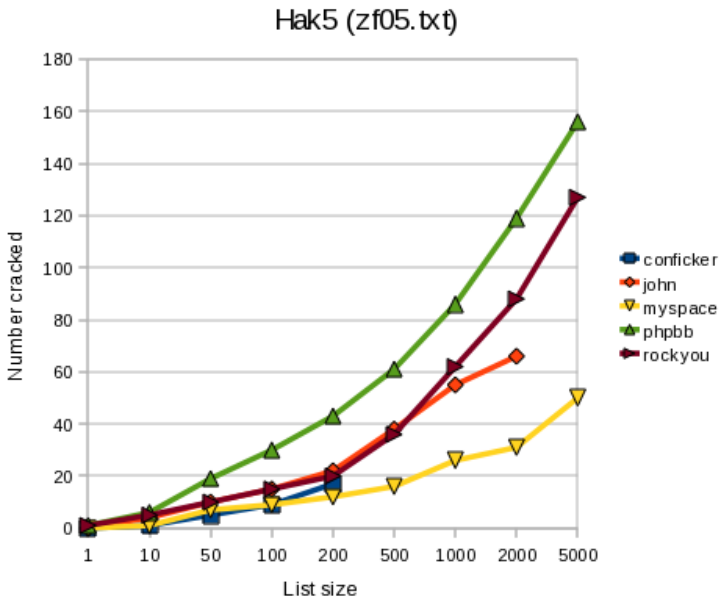

Fig. 3. Hak5 (zf05.txt)

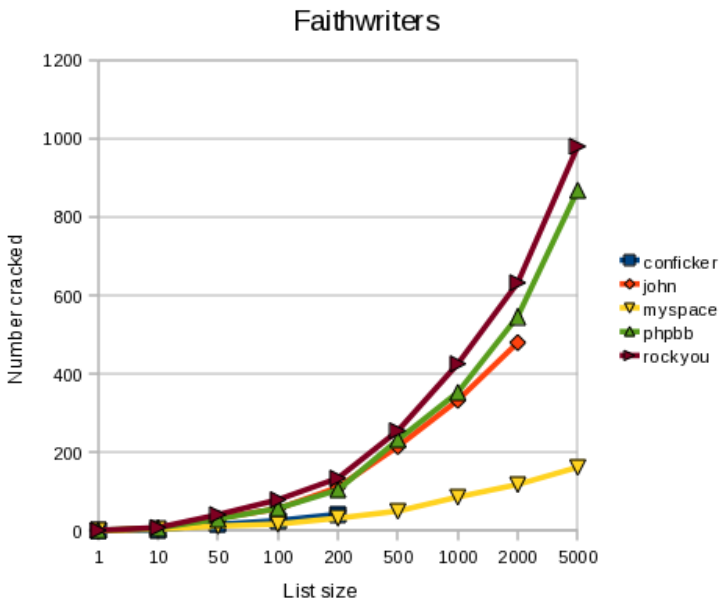

Fig. 4. Faithwriters list

mail list, we can see that the dictionary is winning over the rockyou list, Figure 6. Comparing the sizes of these dictionaries is between the 250MB to less than 90KB.

Above all, comparison of input dictionary list of charts were not made from all dictionaries that were leaked on the Internet. As a result, only the main ones were mentioned, the first leaks, and used to perform a numerous password cracking attacks, with a successful password guesses.

4 TEST

Complementing comparison statistics, next step, 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. Here, 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

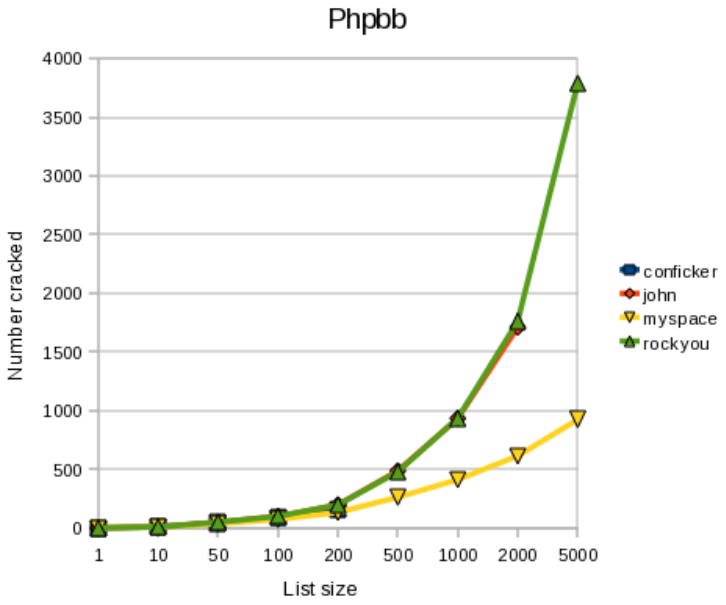

Fig. 5. phpbb list

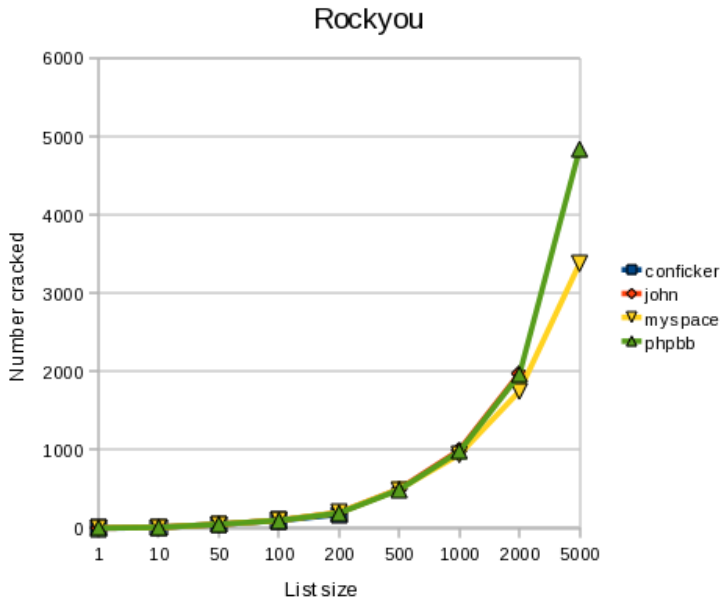

Fig. 6. rockyou list

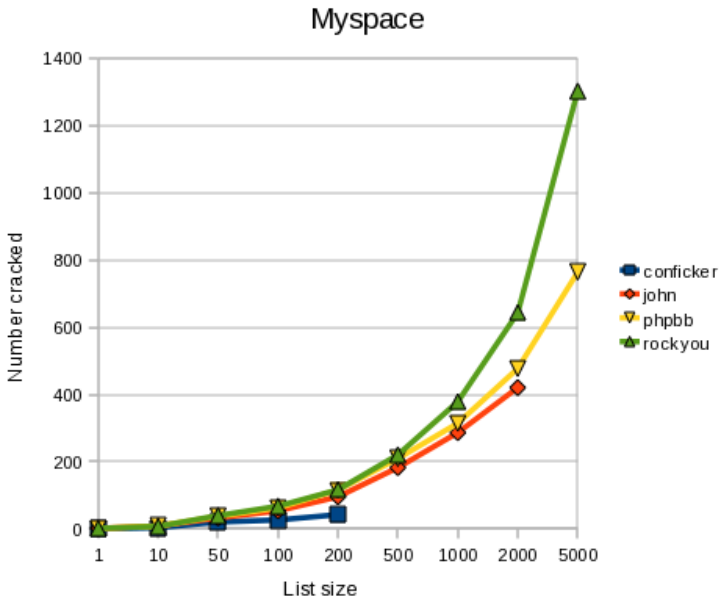

Fig. 7. myspace list

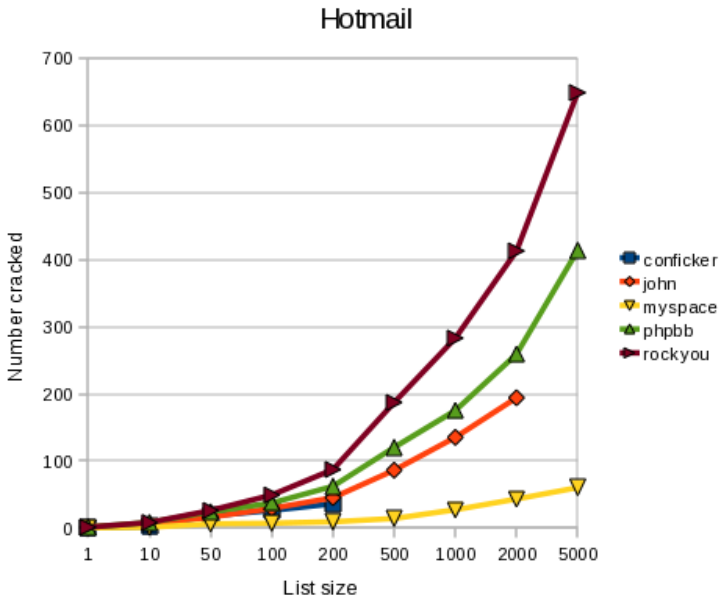

Fig. 8. hotmail list

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, the results came out with successful hit. As a result it still depends on circumference of passwords strength that were used for conducting this tests and the length of input dictionaries.

CONCLUSION

People notoriously are 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.

Indeed, there are many methods and techniques can conduct password cracking, in on-line or offline environment. The tools that can guess the passwords for differential goals. Availability of leaked dictionary from on-line services and generated by tools provide us with help to generate password input dictionaries with 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. It showed examples of leaks and generating password dictionaries, it illustrate already cracked passwords from available input password dictionaries and it presented the test results.

The final aim, is to acquaint the follower about the eventuality methods, tools and input password dictionaries that are available, about the tests that were with an positive cracking password results. To insure for the potential needs of: preventing password cracking, information security audit, password recovery, security policy, 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®), 2011.
2. Travis Altman. Password dictionary generator. http://travisaltman.com/, 2010.

3. Ron Bowes. Passwords, January 2011.
4. Facebook. Facebook user directory. https://www.facebook.com/directory/, 2011.
5. Chris Gates. Tutorial: Rainbow tables and rainbowcrack. Tutorial, 2011.
6. N.S. Gill. The roman military system, 1997.
7. The OpenWall Group. John the ripper password cracker. http://www.openwall.com/, 2010. Openwall Project - Information Security software for open environments.
8. IsNull. Tcbrute, July 2010.
9. JeffXChen. Rainbow table, April 2011.
10. Kestas Chris Kuliukas. How rainbow tables work. kestas.kuliukas.com; Kestas home page, 2006.
11. LLC L0pht Holdings. L0phtcrack password auditor. L0phtCrack Password Auditor, 2009.
12. ElcomSoft Co. Ltd. Elcomsoft products, 2011.
13. Massimiliano Montoro. oxid.it web site. oxid.it web site, 2011.
14. Philippe Oechslin. Making a faster cryptanalytic time-memory trade-off, 2003.
15. Narayanan Vitaly Shmatikov Arvind. Fast dictionary attacks on passwords using timespace. Technical report, The University of Texas at Austin, 2005.
16. L. Stitson. Intro to cryptography notes, 7 2003.
17. Coorporation: TrueCrypt. Truecrypt - free open-source disk encryption - documentation. http://www.truecrypt.org/docs/, March 2011.
18. Russell Dean Vines. Ethical hacking tools and techniques: Password cracking. searchsecuritychannel.techtarget.com, 2007.
19. CHARLES MATTHEW WEIR. Middlechild password cracker. Reusable Security Tools: https://sites.google.com/site/reusablesec/Home/password-cracking-tools/middle-child, 2010.
20. CHARLES MATTHEW WEIR. *Using Probabilistic Techniques To Aid In Password Cracking Attacks*. PhD thesis, The Florida State University, 2010.
21. Matt; Sudhir Aggarwal; Breno de Medeiros; Bill Glodek Weir. Password cracking using probabilistic context-free grammars. Technical report, Internet Security Seminar, 2010.
22. Wikipedia.org. Web crawler. Wikimedia.org;, 2011.