Contemporary Engineering Sciences, Vol. 8, 2015, no. 9, 367 - 374
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ces.2015.5252

An Analysis of CFG Password Against Brute Force Attack for Web Applications

S. Vaithyasubramanian

Sathyabama University, Chennai, India

A. Christy

Sathyabama University, Chennai, India

Abstract

In this paper, we report on a study of brute force attack on CFG passwords. Alphanumeric Passwords are the common and usual mode of authentication for a range of online login. Human proclivities in creating Password draw hackers and enthusiastic Password Crackers to crack down password easily using various techniques, with accessible computing power and available large number of tools. Common attacks on passwords are Brute force attack, Dictionary attack and Hybrid attack. A new method of Alphanumeric Password Authentication for user login is "CFG Password". Context free Grammar Passwords are a class of Alphanumeric Password which differs uniquely from random passwords with certain specifications. CFG passwords are created using the model of the Context Free Grammar. This technique can be used as authentication for web applications. Analysis on CFG Password against Brute force attack is carried out using two open source tools. Comparative analysis has been carried out, based on that suggestions are given to create strong CFG Password for Secured System and where, how it can be used.

Keywords – Password Authentication; Information Security; Alphanumeric Password; CFG Password; Brute force Attack

1 Introduction

Passwords have been used for the purpose of authentication since Roman Periods.

Password-based systems, Validation based on something what the user knows about their identity are generally more common and necessary to access the available resources in World Wide Web through networks and the internet. In this day and age, network load is regularly growing and high-speed infrastructures (1-10Gbps) are becoming increasingly common. The straightforwardness and low cost of execution have contributed to password authentication being the most common and principal authentication method for computer systems. For an individual can be thought to be who he claims to be the point at which the Password imparted between the authenticator and verification is exact. Text based password plays an integral part and vital role in the protection of online identity. A strong password therefore plays an essential role in preventing online identity theft and alike [23, 24].

In hacking and penetration testing one of the most imperative expertises used is the capability to crack user passwords and gain access to system and network resources. Brute force Password cracking is the most familiar method used to crack Passwords [10]. In various ways a brute force attack can happen, however, primarily in an attacker configuring planned values, by creating requests to a server and then analyzing the response. The attacker is able to calculate roughly how long it will take to gain the access of the targeted system based on the number of attempts, estimated efficiency of the system which is used and attacked. An attacker may use a dictionary attack or a traditional brute-force attack with given classes of characters for the sake of efficiency [13, 17].

The confines & Pitfalls in password [20, 21, 22] push hackers and crackers to crack the passwords effortlessly. Passwords are cracked without difficulty by various techniques with the available tools and high speed computers. To overcome from these issue and attacks users has to create strong password which should be a deterrent to attacks. One such innovation in Alphanumeric Password is CFG Password. How to create a CFG Password is explained in the next section.

In this paper, we review the situation and propose a practical, simple, security mechanism. Our system can be incorporated into most web applications. And in this paper we focus specifically on brute force attack on CFG Password. The estimated time to crack CFG Password by brute force attack is calculated with the help of available open source tool. Analysis is done from a sample of 30 CFG Password chosen at random of varying length. Section 2 brief about Brute force attack, Sections 3, 3.1, 3.2 briefs on CFG, generation of CFG password and syntax for generating CFG password with few examples respectively and Section 3.3 summarizes analysis of CFG Password.

2 Brute Force Attack

To hack a website's login page there are many network attacks such as Phishing, tab napping, Social Engineering attack similarly Brute Force attack. Brute force Password cracking methodology is one of the most general, most unswerving Technique used by Password Crackers [10, 11]. Until the correct password is found a brute force attack tries all probable combinations against the

medium. The time span a brute force attack depends on the computer speed, System configuration, speed of internet connection and security features installed on the target system. For Brute force attack estimation time to crack a password is directly proportional to complexity of the password I.e.) if the complexity of the Password increases time taken to crack the password also increase [12, 14]. Using brute force attack tools such as Hydra, attacker can run a large list of possible passwords against various network security protocols until the correct password is discovered. Alike as a way to test the strength of user passwords and overall network integrity hackers and security experts relies on tools. The time consuming and high requirements of processing power are the drawbacks of Brute force attack [15].

3 Context Free Grammar and Language

A Context free grammar [18, 19] G is defined by 4-tuple G = (V, T, P, S) where V is a finite set of non terminal Variable, T is a finite set of terminals, the set of terminals is the alphabet of the language defined by the grammar G. P is a finite relation from V to (V U T)*, called the rules or productions of the grammar G. And S an element of V is the start symbol. Production Rule determines the strings of the language, Strings of the grammar are based on the terminal variables. Context free grammar generates the language called as Context free Language [19]. The Language is a collection of strings generated by means of the production rules defined and using set of terminal, non terminal variables.

3.1 CFG Password

Established methods of Authentication to access the existing immense, wide range of resources in various fields through networks and internet are Alphanumeric Password, Graphical Password and Biometric Authentication. Familiar and usual mode of Authentication is Alphanumeric Password. Password problem gives rise to the development of new Password techniques such as Markov Password [1-4], CFG Password [5] and Array structured Petri Net Password [6, 7]. One of such methodology, strategy for verification for web login is CFG Password; it is a kind of Alpha-Numeric Password. The generation of CFG Password is by utilizing Phrase Structure Grammar [16, 19] of Chomsky Hierarchy. The generations of CFG Password is typical rule based to a certain extent different from random and phrase Passwords. Taking into account the Grammar rule Password can be produced either by user or it can be provided by administrator. Since for a given set of generation rule with input symbols it creates diverse strings of variable length, this kind of Password can be utilized for OTP by service providers for cross check too. Password of varying length and diverse character determination could be possible as per the generation rules. The advantages of CFG Password are it is easy to generate and can be effectively rememberable. From the strings of the language generated using Context Free Grammar, by Choice of the grammar, by choosing the length of the Password, and their desired input symbols a user can choose CFG password or a service provider

can provide a Password for the user. The minimum length of the Password should be 8. User can choose different password for different logins since this grammar generates patterns of strings. By using various combinations and patterns user can set their Password so that it is complex for hackers to crack down their Password. And also users have to keep in mind which prototype of strings they have selected as their password and for which logins.

3.2 Syntax and Illustrations

The Syntax generating few Context Free Grammar is as follows:

1. Grammar generating strings of well balanced Parenthesis: S → S + S | S - S | S * S | S / S | (S) | [S] | {S} | a | b | c | ….| z | 0 |. . .| 9. Where S is the Start Variable and Input Symbols are lower case a to z, Numeric's 0 to 9 and Special Characters ~ to \.
2. Grammar generating strings with special word somewhere on that strings: S → <Letter*> mother <Letter*>; <Letter*> → <Letter> <Letter*> | λ; <Letter> → A | ... | Z | a | ... | z | 0 | ….| 9 | ~ | …| <. Where S is the Start Variable and Input Symbols are lower case a to z, Numeric's 0 to 9 and Special Characters ~ to \.
3. Grammar generating strings with equal number of a's, b's and c's: S → aSBC | aBC; CB → HB; HB → HC; HC → BC; aB → ab; bB → bb; bC → bc; cC → cc. Where S is the Start Variable with Input Symbols a, b and c.

Few examples of CFG Password: (i) (a+b)*(a-b) (ii) {(r-t) + (r*t)} (iii) [x*y]-[x-z] (iv) 123motherabc (v)!@#mother$%^ (vi) aaaaabbbbbccccc (vii) aaaaaacbbbbbb (viii) xyxycyxyx (ix) abxyyxba (x) yxbaabxy.

3.3 Estimation Time to CRACK CFG Password

Estimation time to crack CFG password by Brute force attack are tabulated below with figures showing seconds to crack CFG Password. 30 random passwords generated using CFG structure is considered for the analysis.

For CFG Password						
In Seconds	Standard Desktop PC	Fast Desktop PC	GPU	Fast GPU	Parallel GPUs	Medium size botnet
Average	20049372105214600	5275316691235710	2110127518281570	991877325379344	99187711186076	19994498613
Maximum	599581594000000000	157784630000000000	63113852000000000	29663510440000000	2966351040000000	599582000000
Minimum	0	0	0	0	0	0
For Common Password						
Average	1070	270	102	55	5	0
Maximum	28800	7200	2820	1440	120	0
Minimum	0	0	0	0	0	0

Table: 1 Attack estimation time estimated by Password -checker. Online-domain-tools [9].

For CFG Password				
In Seconds	ZX Spectrum The popular home computer from the 80s	*Mac Book Pro (2012) Popular laptop with powerful Intel Core i7 CPU	Conficker botnet One of the most prolific botnet	Tianhe-2 Supercomputer The world's fastest supercomputer
Average	3178489485387	1105082793951	1051947323320	1051900277375
Maximum	31557000000000	31557000000000	31557000000000	31557000000000
Minimum	7200	4	1	1
For Common Password				
Average	3	1	1	1
Maximum	4	1	1	1
Minimum	2	1	1	1

Table: 2 Attack estimation time estimated by blog.kaspersky.com/Password –check [8].

Comparison between the common password and CFG password are tabulated which shows that it will take time to crack CFG password is than common Password. This suggests that CFG Password can be effectively used for web login Authentication.

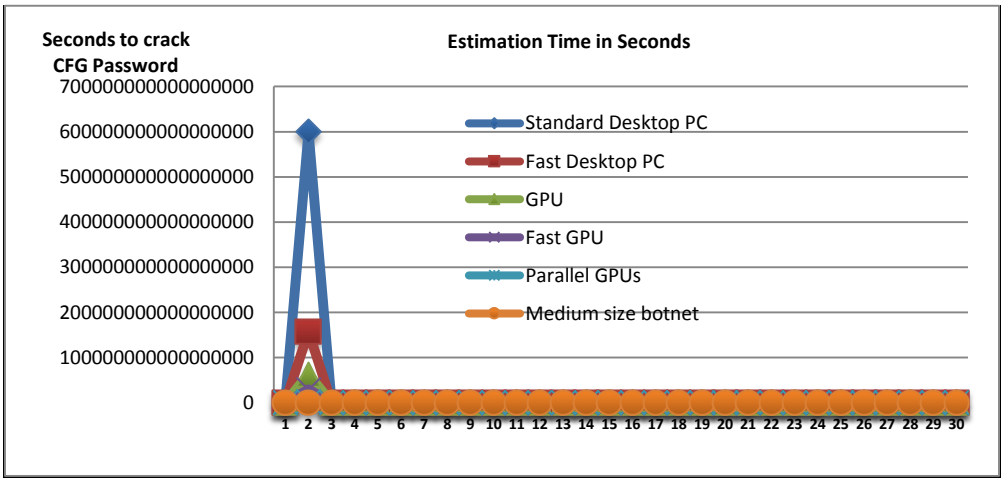

Fig 1. Attack estimation time estimated by Password -checker. Online-domain-tools [9].

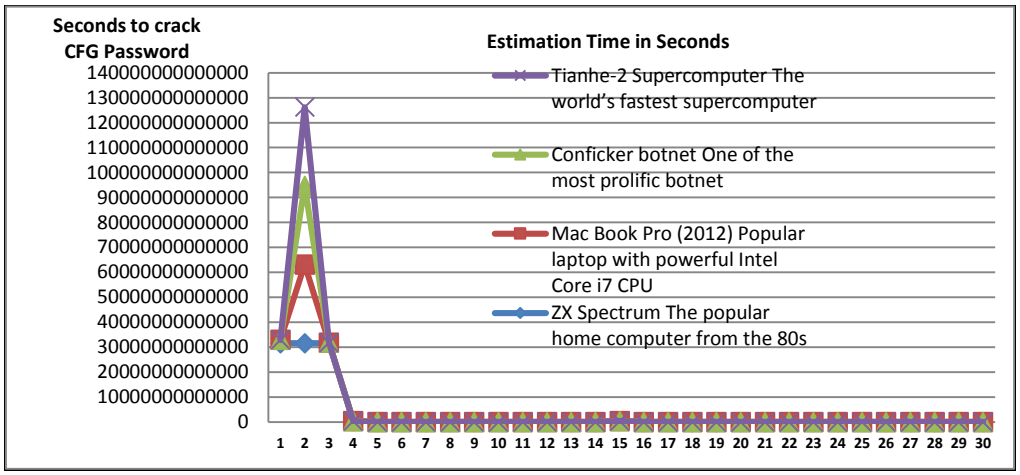

Fig 2. Attack estimation time estimated by blog.kaspersky.com/Password –check [8].

5 Conclusion

The brute force attack is difficult to deal with the Passwords of complex and unpredictable length. From the tables 1 and 2 showing the average, upper limit and minimum estimation time to crack Password it is clear that to a larger extent for CFG Password it will require more time to crack than the common Password. Time to crack CFG Password like mathematical expression with well balanced parenthesis and passwords of length more than 10 shows good resistance against brute force attack. This paves a new path for effective data and network security. Good resistance towards brute force attack shows CFG Password can be effectively used as Password authentication for network applications. Here the tool can be built to generate CFG password with predefined set of rules and functions which would be sequential executed for each user instruction to generate potential CFG Passwords. This tool supports admin group also for password setup, resets, random password requirements and OTP. To enhance and enrich the performance of this CFG Password technique high level of user support and research would lend a hand.

References

[1] S. Vaithyasubramanian, A. Christy "A Scheme to Create Secured Random Password Using Markov Chain" Advances in Intelligent Systems and Computing, Springer India, Vol. 325, pp 809-814, 2015.
http://dx.doi.org/10.1007/978-81-322-2135-7_85

[2] S. Vaithyasubramanian, A. Christy, D. Saravanan "An Analysis of Markov Password against Brute Force Attack for Effective Web Applications" Applied Mathematical Sciences, Vol. 8, no. 117, pp5823 – 5830, 2014.
http://dx.doi.org/10.12988/ams.2014.47579

[3] S. Vaithyasubramanian, A. Christy "An Analysis on 1-Step Transition Probability Matrix and 2-Step Transition Probability Matrix of Markov Passwords" International Journal of Applied Engineering research, Vol. 9, Number 20, pp7745-7753, 2014.

[4] S. Vaithyasubramanian, A. Christy "A study on Markov chain password using Bayesian inference" CiiT International Journal of Artificial Intelligent Systems and Machine Learning, Vol. 3, No 3, 2014.

[5] S. Vaithyasubramanian, A. Christy "A practice to create user friendly secured password using CFG" International Conference on Mathematics & Engineering Sciences, Chitkara University, Punjab, pp39, March 2014.

[6] S. Vaithyasubramanian, A. Christy, D.Lalitha "Generation of Array Passwords Using Petri Net for Effective Network and Information Security" Advances in Intelligent Systems and Computing, Springer India, Vol.1, pp189 – 200, July 2014. http://dx.doi.org/10.1007/978-81-322-2012-1_20

[7] S. Vaithyasubramanian, A. Christy, D.Lalitha "Two factor Authentication for Secured Login Using Array Password Engender by Petri net" Accepted for Procedia Computer Science, Elsevier, 2015.

[8] http://blog.kaspersky.com/Password -check/

[9] http://Password-checker. Online-domain-tools.com/

[10] http://www.howtogeek.com/166832/brute-force-attacks-explained-how-all-encryption-is-vulnerable/

[11] Carlisle Adams, Guy-Vincent Jourdan, Jean-Pierre Levac and Francois Prevost "Lightweight Protection against brute force login attacks on web applications" PST, pp181 – 188, IEEE – 2010.
http://dx.doi.org/10.1109/pst.2010.5593241

[12] Jim Owens and Jeanna Matthews "A Study of Passwords and Methods Used in Brute force SSH attack" In USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), 2008.

[13] Mudassar Raza, Muhammad Iqbal et.al "A survey of Password Attacks and Comparative analysis on methods for secure Authentication" World Applied Science Journal 19(4): pp439 – 444, 2012.

[14] Neeraj Kumar "Investigations in Brute force attack on Cellular Security Based on Des and Aes" International journal of Computational Engineering & Management, Vol 14, pp50 – 52, October 2011.

[15] Richard Clayton "Brute force attack on cryptographic keys"- file:////H:/brute force attack / brute.html, Oct 2001.

[16] Akshay Kanwar, Aditi Khazanchi, Lovenish Saluja "Analyzing Ambiguity of Context-Free Grammars" International Journal of Engineering and Computer Science, Volume 2 Issue 10, pp2921-2926, October, 2013.

[17] Mansour Alsaleh, Mohammad Mannan, and P.C. van Oorschot "Revisiting Defenses against Large-Scale Online Password Guessing Attacks" IEEE Transactions on Dependable and Secure Computing, vol. 9, no. 1, pp128-141 jan/feb 2012. http://dx.doi.org/10.1109/tdsc.2011.24

[18] Theodore Norvell "A Short Introduction to Regular Expressions and Context Free Grammars" Software Engineering 7893, November 8, 2002.

[19] Hop croft, John E.; Ullman, Jeffrey D. (1979), Introduction to Automata Theory, Languages, and Computation, Addison-Wesley. Chapter 4: Context-Free Grammars, pp77–106.

[20] Sarah Granger, "The Simplest Security: A Guide to Better Password Practices" - http://www.symantec.com/connect/articles, July 2011.

[21] Jeff Yan, Alan Blackwell, Ross Anderson, Alasdair Grant "Password Memorability and Security: Empirical Results" IEEE security & privacy Volume: 2, Issue: 5, pp25 – 31, 2004. http://dx.doi.org/10.1109/msp.2004.81

[22] Edward F. Gehringer "Choosing passwords: Security and Human factors" IEEE international symposium on Technology and Society, pp369 – 373, 2002. http://dx.doi.org/10.1109/istas.2002.1013839

[23] Dinei Florencio, Cormac Herley, Baris Coskun "Do strong Web passwords Accomplish Anything?" Proceedings of the 2nd USENIX workshop on Hot topics in security, ACM Digital Library, 2007.

[24] Dinei Florencio, Cormac Herley " A Large-Scale Study of Web Password Habits" Proceedings of the 16th international conference on the World Wide Web, ACM Digital Library, pp657-666, 2007.
http://dx.doi.org/10.1145/1242572.1242661

Received: March 2, 2015; Published: March 23, 2015