Implementation of a Secured Authentication System using a Policy Generator with Email Notifications

Nikhil Chakravarthy Mallela, Arun krishna Chitturi, Swarnalatha Purushotham

Abstract: For securing the login, passwords of users from intruders and hackers, the website owners and administrators are providing certain guidelines to the users to create secure and strong passwords using a mechanism called Password Checkers. These guidelines which are provided helps the users to create strong passwords, these guidelines are also becoming the raw input for the hackers as they clearly show based on which policy the password was generated which increases the risk for brute force attacking with more ease. There by increasing the success rate probability for the brute force attackers. To overcome and to decrease the success probability for brute force attacking the Dynamic Password Policy Generator is being devised.The profiles of users are built and maintained by the system automatically bases on the interaction with the monitored database in training phase. This DBSAFE system will help both the administrator as well as the users to feel secured in terms with their data security. Also whenever, an unsuccessful attempts leaving a notification through an email will always add a extra layer of security to the system. When the system's critical files were all under watch and someone try to access those, concerned people will be intimated to verify the system security keeping the system and database safe and healthy.

Keywords : Randomized structure generation, User favored password, Password monitor, Passwords anomalies

I. INTRODUCTION

To shield password clients from making straightforward and normal passwords, real sites and applications give a password quality measure, in particular a password checker but unfortunately such static checkers spill the data and enabling the intruder to improve the execution of their attacks[3]. For solving this problem, we propose and devise the Dynamic Password Policy Generator, to be specific DPPG, to be a compelling and usable other option to the current password quality checker. DPPG is a diversity-based and database-aware application that generates password creation policies dynamically for the users. Instead of purely focusing on the complexity of candidate passwords, DPPG enforces a baseline complexity on the passwords like more than 6 characters long to protect them from simple attacks, e.g., dictionary, brute-forcing[1]. More focus is put on protecting the password distribution within a database by preventing aggregation of similar passwords that form a characteristically biased distribution. As long as a candidate password meets the policy, it is accepted and no additional strength feedback is provided. The policies are generated to search for candidate passwords that balance the password characteristics distribution. To make the system more complex for the hackers to attack the database, the database will also be subjected to the anomaly detection which helps the user even when the intruder attempts to get access to the database it'll help them in capturing the respective person with the help of the alerts received by the administrator [4]. This is specifically designed for the RDBMS which is mostly used in the current cyber world, having the system uses predetermined policies guiding automated and/or human response to the anomaly. Password authentication is used mostly because of less implementation expenditure and ease to change [5]. Another solution for password security was TarGuess, a framework for characterizing specifically chosen scenarios [7]. Two state of the art techniques like PCFG, and Markov based ones are also used [8]. Detnom an anomaly detection mechanism is also used to overcome password insecurity [9]. Some password meters are used to measure the strength of password labels [12].

Manuscript published on 30 December 2019.
***** Correspondence Author (s)

Nikhil Chakravarthy Mallela, Mtech Integrated Software Engineering, Vellore Institute Of Technology, Vellore, India. Email: nikhilchakravarthy.mallela@gmail.com

Arun krishna Chitturi, Mtech Integrated Software Engineering, Vellore Institute of Technology, Vellore, India, chitturiarunkrishna@gmail.com

Swarnalatha Purushotham*, Computer Science, Vellore Institute Of Technology, Vellore, India. Email: hgswarna@gmail.com

II. LITERAURE SURVEY

A. Dynamic password policy generation system

Method - A part of aimlessly selected passwords are used as training data and the rest are used as target data. All this data is taken from a leaked set of passwords. A threat model is also developed [2]. Let us consider an attacker tries to breach a bundle of password hashes which is leaked from a website using a strength checker for passwords. They developed the DPPG (Dynamic Password Policy Generator). This generates the dynamic password policies for the users[2]. Advantages - Simple and intuitive way rather than searching with complicated models.We judge the effect of exploiting present economic password strength checkers from attacker's view as passwords of same strengths will almost have the same pattern using which attackers can hack [2]. In DPPG each user will be getting unique passwords created by server and hence the user does not know what policy the other user gets unlike conventional method [2].

Retrieval Number: A10361291S319/2019©BEIESP
DOI:10.35940/ijeat.A1036.1291S319
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

B. Generating pronounceable security passwords

Method - The "Sandia System" is a pronounceable password generator which uses the next following methodology [13]. 25 different templates are created. Each template resembles the words formed by a vowel, constants, or vowel followed by a consonant followed by a vowel, etc…have been created. In order to create a password the system explicitly takes any one from twenty five templates. Then the system chooses a seven digits long password from previously chosen template [13]. Then at last for security either a digit from 0 to 9 is added or any alphabet is added to any of the eight positions randomly to increase the security. The users are given distinct such passwords and are allowed to choose any one password.

Advantages - The attacker needs to do exhaustive search which provides more security to find the pronounceable passwords in the large buckets (Inside the buckets all the word templates are kept), these passwords which provides greater security than conventional systems and techniques [13].

C. Administrative password generation

Method - The method includes obtaining a tag associated with a client computer, and generating a password using the tag. The password is used for an application accessible by the client computer [11]. The method includes updating the application with the password to allow access to the Application via the generated password. The tag may be Stored in a database or other data store. The password for the application is generated using the tag. A system for password generation and control is provided. The system includes a first computer and a second computer [11]. A password component obtains a tag and uses the tag to generate a password for an application used on the first computer. The tag is related to the first computer. A database stores information including the tag. A manager component is operable on the second computer to obtain the tag from the database and uses the tag to generate the password to enable access to the application on the first computer [11].

Advantages - In an organisation, information security is very important which can be achieved by employing passwords that are used by a user to gain access to the computer or organisation services. The passwords generated are encrypted and hence provides much security to the computers [11].

D. Password Management with Storage Optimized Honeyword Generation

Method – The Paired Distance Protocol (PDP) uses three data (a) Username (b) Password and (c) a Random String RS of any number or alphabet. length of RS is set to 3 as default length. Using the honey circular list the honeywords are generated and kept in the password record [1]. A list Wi is kept up with the honeyword age approach against each username Ui. Another document is utilized to keep up the list of the right password in an alternate framework in order to fake it out. And when the original list Wi is replaced with honeywords in order to confuse the attacker. If the intruder enters the honey words and gets confused and by chance if it is right the honey checker will give the positive result [1].

Advantages – The risk of brute force can be solved by identifying the password splitting with the Honey word based confirmation protocol. Hacks can be identified faster with honey words [1].

E. Dynamic password authentication

Method – This method uses a grid which consists of alphabets, special characters and numericals. Every character is used at least two times in each grid. The password is generated from the grid. This password differs every time when the page is refreshed [10]. A specific pattern is set for every user when the user signs up for the first time. Character are selected from the grid in predefined order by the user[10].

Advantages - It is easy to use, although one might think that memorizing a collection of boxes is very difficult, but on the opposite side this method is simple [10]. User is allowed to choose a set of boxes in such a way that a pattern is formed with them. The size of the pattern can be of user's choice. Therefore it will be very easy for the user to remember the password. The whole system is dynamic. A unique password is generated every time when the user logs in. The major asset of this system is that, when a hacker cracks the password, he/she will have access to that particular account until the password is changed [10]. Therefore the risk of identifying or figuring out the pattern is very less. In this way the accounts cannot be compromised easily. This method is economical in terms of cost because it uses simple algorithm which analyses the security in a very simple way.

F. Password security system with two way authentication

Method - The system asks the new user to give a password, an image and a secret key and an uncommon username. The algorithm recognizes two sets of pixels on the image. One set of pixels and the key are utilized in image verification procedure [6]. The other set of pixels is used to generate the password derivative. The key is secured in user's database. The difference in positive and negative signs of user's key is stored as secret key derivative. The difference in positive and negative signs of user's text password is stored as unique segment of password derivative [6].

Advantages - This system comes with two layers of security. Therefore it is more secured and opposes unauthorized and unauthenticated login attempts. Image authentication layer gives security and prevents most common attacks like keystroke recording, eaves dropping and hidden cameras. Moreover the user's passwords are not stored anywhere. They are derived from the image that user provides during registration [6]. Therefore the database does not contain any password.

G. Design of a new web database security model

Method - The older web database system was tri-hierarchy model. A twice login module has two connections with web database [14]. The former one connects with web database and the latter logins web database with the right account. Audit module monitors, records and controls the user's activity in the system. In order to assure security of the database, the audit system evaluates the logs and log information in an easy and coherent way [14].

Advantages - The audit module helps to offer multiple audit selections for system auditor administrator [14]. Program Control Modules. Database Rights Control. It consists of a virtual table named VIEW, it has all columns and rows of data with their names. But view don't store true data in the database, columns and rows data of view are derived from tables and view could be produced dynamically when application program cited it. Other Security Measures [14].

H. Real time password generation apparatus

Method - There is a non-transitory computer readable storage medium which has stored instructions for generating a password in real time, comprising a first code segment, to create at least single password map while creating the account associated with the user, a second code segment, to generate and provide a random password sequence grid to the user, and third code segment, for authenticating the user for accessing the account using a password created by the user, wherein the password is created by the user using the random password hint sequence grid and the at least one password map [15].

Advantages - To gain access to Secure computer systems, bank accounts, and other processes within a computer or Internet device [15]. They can provide session based authentication between a user device and External system, Such as a web service on the Internet [15].

III. EXISTING MODEL

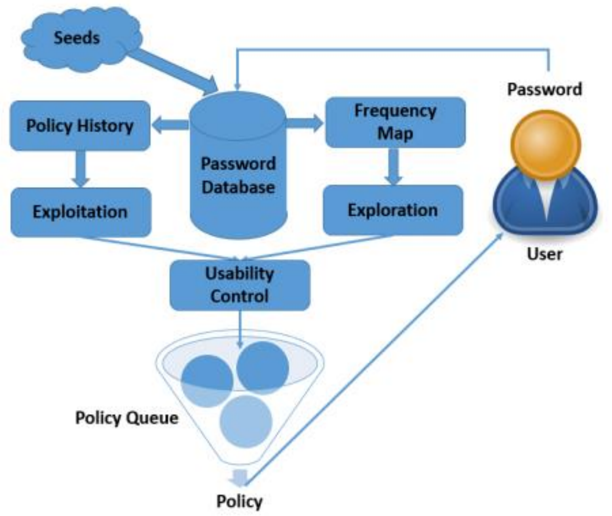


Fig 1- Existing DPPG Model

Firstly the system administrators can induce explicit passwords which work as seeds in the database. The seeds can form a list certain desired password characteristics to insert in the generated password. Now DPPG starts to generate password policies on the basis of these seeds. Users can type their desired password characters which will also be stored in the password database. Also the user will be given which are all the good and bad characters which has to be used and avoided for generating the random password. For generating the password policies intelligently, DPPG maintains a global characteristics frequency map and a history of generated password policies that can approximate the current password distribution.[2]

There are two different ways for DPPG to enlarge the usable space of passwords. The exploitation mode balances password distribution with the help of password history policy. The exploration mode enlarges the usable space of passwords by proposing different characteristics on the frequency map. Before an incoming password is hashed, DPPG extracts its characteristics and stores the metadata in the frequency map, which keeps tracks of the overall distribution of password attributes. Then based on the dynamic password policy it will generate a random password containing both seeds and the user's desired password characteristics which cannot be identified easily by any hacker. Each time a new password will be generated as every single time the policy changes.

IV. PROPOSED MODEL

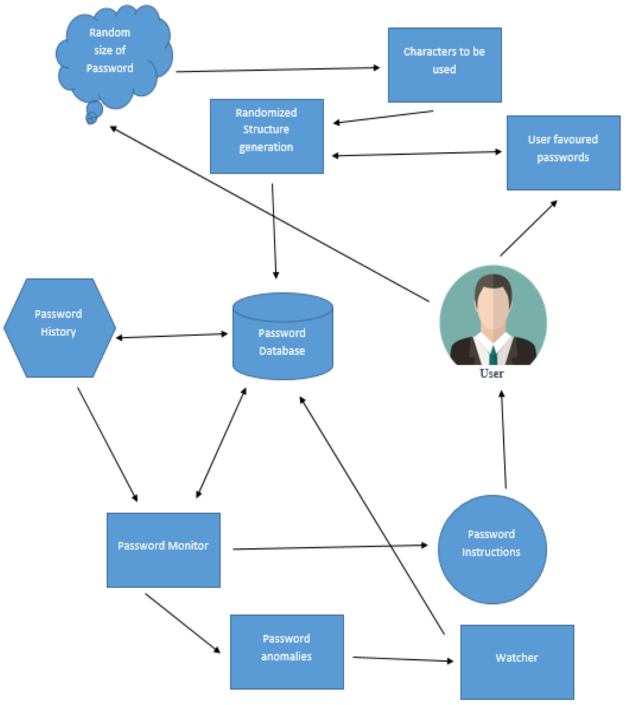


Fig 2 – Proposed Model

According to this proposed model, the password can be viewed as a combination of the user favoured password and the system favoured password. Here in this model the user favoured password is the part of the password which is set completely according to the user will, wherein the system favoured password is up to the system.

Hence the system favoured password does involve certain mechanism to generate the password, the mechanism can be described as follows: Initially the system generates a range for the length of the password from which user selects a number "n" (which will be the entire password length). After which depending on the "n" the system generates a set of characters (which can be either alpha-numeric or special characters). On the other hand, the number "n" will be processed to get divided by a number from the set of listed prime numbers. The floor or ceil function of the output of the prime number division of the number "n" (say "s") will be used to extract the random substring of length "s" from the generated structure using characters generated by the system.

This extracted substring, system favoured password along with user favoured password forms the entire password which can be further encrypted with the encryption algorithms before feeding it to the database. Here when the length of the password is “n” and output of the either floor or ceil function is assumed as “c”, the structure generated will be of the size “c” and the user favoured password will be of the size “n-c”. This will be form “n” length password at the end. The user can have the user favoured password at the end or at the beginning, the only condition is that user cannot split the system favoured by any chance which will result in the invalid password. The system will not accept passwords of such a type. The password which followed these sequence of steps will only be accepted by the system. The so generated or accepted passwords will be stored in the database and will be subjected to the evaluation with existing passwords to eradicate the duplications or same structured passwords. All the process will be monitored by the password monitor (flag) and on the successful completion of all the steps the instructions will again be fed to the user for the approval and will be fed to the database.

For example, let’s consider the scenario of a random user named Bob

Bob: Selects the length of the password as 9

System: Generated some random characters – M, N & C

Bob: can places this M, N & C as MNCM or combinations of MNCM (like CNMM, MCNM etc.) in the password at any position (either in the beginning or ending or in the middle), since floor of the number 9 when divided by the first prime number is 4. Bob can have the remaining five letters in the password can be of his choice.

Bob: Sets the password as Bob99MMCN

With help of this model, the password can be kept secured the user just have to keep note of the structure that was opted and the user favoured password. As the password is a combination of the system favoured password and the user favoured password, half of the password which is a randomized structure and the remaining half user willed and the structure can be at any position in the final password. This makes difficult for the attackers or hackers to guess the password which adds a kind of another layer of security to the password or database.

Whenever a password anomaly like trying to guess the password with combinations for more than twice will initiate the watcher, recording the instance into the database which helps in tracking the password and the system. The watcher keeps the system at benefit sending with notification at the very least time once configured properly. This model helps in achieving a very good state of security to the passwords as well as for the systems.

V. RESULTS AND ANALYSIS

A. Complete Analysis

Initially let us begin with recognizing the problems with present password strength checkers and figure out in an adversarial perspective. We have identified that password strength checkers are serving attackers in launching serious and powerful attacks. The main reason for this is that they rely on static scoring policies. These policies apply partiality in password distribution. These checkers give an advantage to the hackers to choose the training data which is similar to the target passwords. In order to avoid this problem DPPG is implemented to generate dynamic password policies based on diversity metric and the current password distribution. By our knowledge, DPPG is the first policy generator which gives unpredictable policies. Anyone may debate that the potential solution to password checkers is to have powerful, advanced and better website technologies to cover the policies and recognize the harmless password queries. Nonetheless, it results lagging in strength feedback. There may be high false-positive rate in detection.

This proposed model highly advantageous in many terms, the randomization of the password length from a system generated range makes it difficult to find the length as well as for brute forcing. The random characters generated for the creation of the password have not been restricted as good and bad characters, reducing the restrictions on the user to choose the password freely. This gives the users more ease to remember the passwords and to handle them. The part of random substring completes the process of generating a secure system favoured password, extracting a substructure from a very large structure adds a layer of security from the people those were well associated with pattern of the user password. The positioning of this structure at various locations like at the beginning, end and middle keeps hard for guessing. The password anomaly concept gets triggered when someone tries to mimic or guess the password of the user by which the watcher gets initiated and the database administrator and the user gets updated. The system email notification facility provided by the watcher helps in sending a genuine report to the administrator and the user adding a high level security to the database and the system.

All the anomalies like the brute forcing on the system, the dictionary attack and other trial and error method attacks can be identified and reported to the administrator. This helps in keeping the database healthy and secure.

B. Watcher

The Watcher is a watchtower for linux, it monitors user configured system file integrity and access, sudo command access, ssh logins, iptables changes, it emails the admin on any of this events using postfix and also logs the incidents and will also dump system memory contents to a file for forensics. We also run a rootkit scan on installation and email the results to the admin. We also made a systemd service for the same.

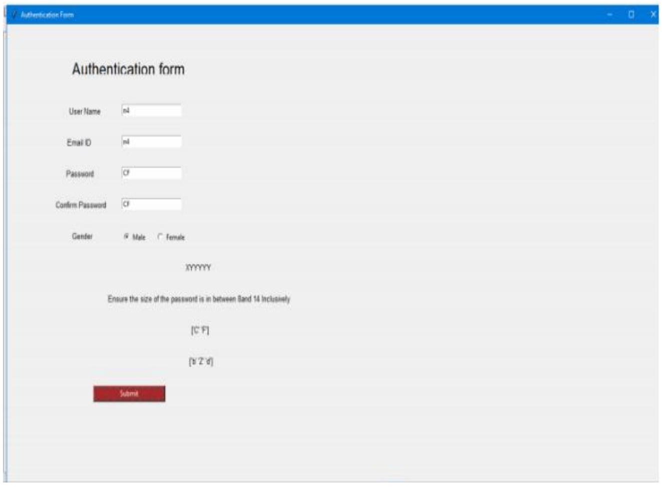
- The key features of The Watcher are
- File Integrity Monitoring
- File Access Monitoring
- IPTABLES rule monitoring
- Sudo access monitoring
- Email alerts to specified admin
- Dump system memory to file on incident
- Logs for analysis
- Rootkit detection on install

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

When it's first run it creates a baseline for normal system resources, it stores the hashes in the work_dir folder then it continuously monitors the system for changes and file accesses. It also monitors common system resources such as the iptables and sudo auth and ssh logs.

C. Implementation results

The whole system is dynamic. Every time when the user logs in a new password is generated. The major advantage is that, usually once a hacker gets a password he has access to the account till the password is changed. Even if the hackers manage to find the password, the chances of identifying the pattern is very less. In this way the data is not compromised or leaked. The proposed idea takes care of the security of the system and also it takes care of authentication. This is very economical in terms of cost, the reason is that it uses a simple algorithm to check authentication. It is easy to use, although one might think that memorizing a collection of boxes is very difficult, but on the opposite side this method is simple. User is allowed to choose a set of boxes in such a way that a pattern is formed with them. The size of the pattern can be of user’s choice. Therefore it will be very easy for the user to remember the password. Thus the comparative results for the dynamic password policy generator was given


Authentication Form

Authentication form

User Name n4

Email ID n4

Password CF

Confirm Password CF

Gender ● Male ○ Female

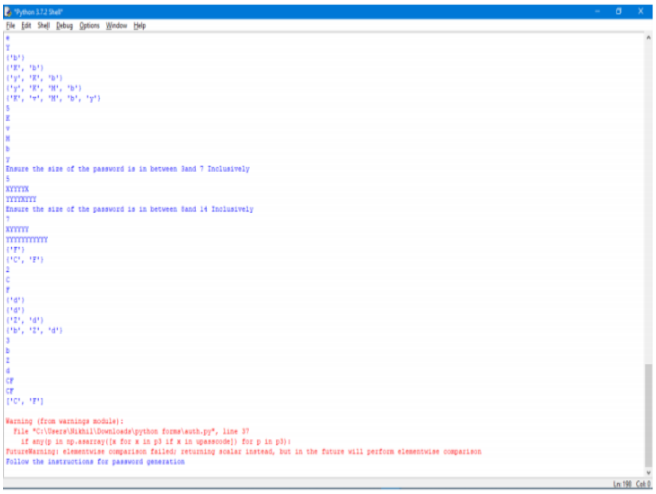
XYYYYY

Ensure the size of the password is in between 8and 14 Inclusively

['C' 'F']

['b' 'Z' 'd']

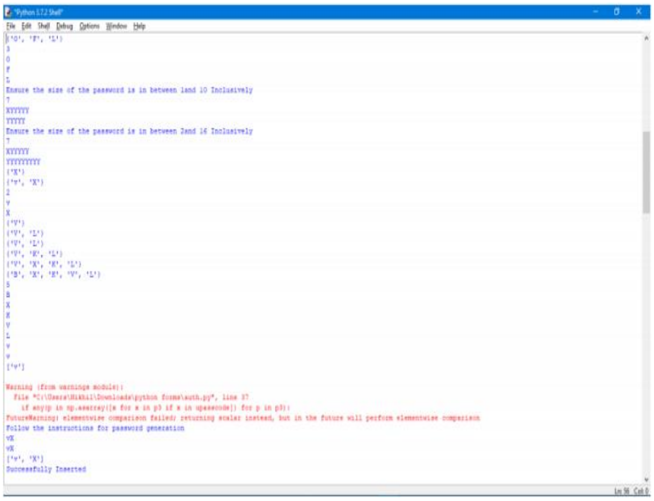
Fig 3 – Authentication form of the proposed system



```
e
Y
{'b'}
{'K', 'b'}
{'y', 'K', 'b'}
{'y', 'K', 'M', 'b'}
{'K', 'v', 'M', 'b', 'y'}
5
K
v
M
b
y
Ensure the size of the password is in between 3and 7 Inclusively
5
XYYYYX
YYYYXYYY
Ensure the size of the password is in between 8and 14 Inclusively
7
XYYYYY
YYYYYYYYYYY
{'F'}
{'C', 'F'}
2
C
F
{'d'}
{'d'}
{'Z', 'd'}
{'b', 'Z', 'd'}
3
b
Z
d
CF
CF
['C', 'F']

Warning (from warnings module):
  File "C:\Users\Nikhil\Downloads\python forms\auth.py", line 37
    if any(p in np.asarray([x for x in p3 if x in upasscode]) for p in p3):
FutureWarning: elementwise comparison failed; returning scalar instead, but in the future will perform elementwise comparison
Follow the instructions for password generation
```

Fig 4 – Implementation of proposed system



```
{'O', 'F', 'L'}
3
O
F
L
Ensure the size of the password is in between 1and 10 Inclusively
7
XYYYYY
YYYYY
Ensure the size of the password is in between 2and 16 Inclusively
7
XYYYYY
YYYYYYYYY
{'X'}
{'v', 'X'}
2
v
X
{'V'}
{'V', 'L'}
{'V', 'L'}
{'V', 'K', 'L'}
{'V', 'X', 'K', 'L'}
{'B', 'X', 'K', 'V', 'L'}
5
B
X
K
V
L
v
v
['v']

Warning (from warnings module):
  File "C:\Users\Nikhil\Downloads\python forms\auth.py", line 37
    if any(p in np.asarray([x for x in p3 if x in upasscode]) for p in p3):
FutureWarning: elementwise comparison failed; returning scalar instead, but in the future will perform elementwise comparison
Follow the instructions for password generation
vX
vX
['v', 'X']
Successfully Inserted
```

Fig 4 – Implementation of proposed system

VI. CONCLUSION

The proposed model is an alternative to the DPPG with email notification on successful and unsuccessful, suspicious activity. Traditional password strength checkers reveal the password policies to everyone which makes the very complex passwords also vulnerable to Brute force attack and other capable attacks. DPPG generated passwords though have a good memorisable rate, the user gets restricted in choosing passwords and not easy at every instance of password. People can argue that a potential solution to the password checker limitations is to have better web technologies to hide the policies and detect malignant password strength querying. However, it can result in delay in strength feedback and high false-positive rates in detection. Further, it does not resolve the fundamental bias in password distribution. The proposed model in this paper, generates dynamic structures with random characters and provided to users each time through which the passwords are highly safe as the policies are hidden and not visible to the hacker thereby reducing the risks. This model offers many layers of security to the process of creating password and more ease to the user in remembering the password. As the traps and the dynamic nature of the authentication system is very hard to penetrate. Having the feature of emails for the tracking of suspicious activities on a system using the trap files keeping them under watch by The Watcher helps the admin to have a safe and healthy system. Using The Watcher setting traps as well as the securing the critical resources will be under a safe hawk eye to ensure the security.

REFERENCES

1. Sivaji, N., & Yuvaraj, K. S. (2018). Improving Usability of Password Management with Storage Optimized Honeyword Generation.
2. Yang, S., Ji, S., & Beyah, R. (2017). DPPG: A Dynamic Password Policy Generation System. IEEE Transactions on Information Forensics and Security, 13(3), 545-558.
3. Zhang, S., Zeng, J., & Zhang, Z. (2017, October). Password guessing time based on guessing entropy and long-tailed password distribution in the large-scale password dataset. In 2017 11th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID) (pp. 6-10). IEEE.

4. Sallam, A., Bertino, E., Hussain, S. R., Landers, D., Lefler, R. M., & Steiner, D. (2015). DBSAFE—an anomaly detection system to protect databases from exfiltration attempts. IEEE Systems Journal, 11(2), 483-493.
5. Zheng, Wantong, and Chunfu Jia. "CombinedPWD: A New Password Authentication Mechanism Using Separators Between Keystrokes." 2017 13th International Conference on Computational Intelligence and Security (CIS). IEEE, 2017.
6. Biswas, Subhradeep, and Sudipa Biswas. "Password security system with 2-way authentication." 2017 Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN). IEEE, 2017.
7. Wang, Ding, et al. "Targeted online password guessing: An underestimated threat." Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. ACM, 2016.
8. Wang, Ding, et al. "Understanding passwords of chinese users: characteristics, security and implications." CACR Report, Presented at ChinaCrypt (2015).
9. Hussain, Syed Rafiul, Asmaa M. Sallam, and Elisa Bertino. "Detanom: Detecting anomalous database transactions by insiders." Proceedings of the 5th ACM Conference on Data and Application Security and Privacy. ACM, 2015.
10. 10. Balaji, R., and V. Roopak. "DPASS—Dynamic password authentication and security system using grid analysis." 2011 3rd International Conference on Electronics Computer Technology. Vol. 2. IEEE, 2011
11. Freeburne, Alexander B. "System and method of enterprise administrative password generation and control." U.S. Patent No. 8,775,820. 8 Jul. 2014.
12. De Carnavalet, Xavier De Carné, and Mohammad Mannan. "From Very Weak to Very Strong: Analyzing Password-Strength Meters." NDSS. Vol. 14. 2014.
13. Ganesan, Ravi. "Method and system for generating pronounceable security passwords." U.S. Patent No. 5,588,056. 24 Dec. 1996.
14. Yangqing, Zhu, et al. "Design of a new web database security model." 2009 Second International Symposium on Electronic Commerce and Security. Vol. 1. IEEE, 2009.
15. Bodavula, Vikram. "Real time password generation apparatus and method." U.S. Patent No. 8,984,599. 17 Mar. 2015.

AUTHORS PROFILE

Nikhil Chakravarthy Mallela is a tech enthusiastic whose is pursuing his 4th year M.Tech (S.E) in Vellore Institute of Technology, currently working in the domains of Machine Learning and Artificial Intelligence. He strongly believes in understanding the of the core concepts will always help in finding an optimal solution for an unpredictable problems. His dream is to work for the defense sector in India, which indirectly helps in serving his nation without stepping into the battle field.

Arun krishna Chitturi is a student who is very much interested in exploring and implementing emerging trends in IT sector. He is currently pursuing his 4th year M.Tech (S.E) in Vellore Institute of Technology, Vellore. He has attended two international conferences, 8th International Conference Soft Computing for Problem Solving at VIT Vellore and The 2nd World Summit on Advances in Science, Engineering and Technology at Indiana University-Purdue University, Indianapolis, USA. He also published a paper in International Journal of Advanced Trends in Computer Science and Engineering in Natural Language Processing domain. He is interested to work in cybersecurity, block chain, Deep learning and Natural Language Processing domains along with other domains in IT. He also holds the current board position of Technical Director (CSE) of IETE - ISF, Best chapter of VIT 2018-19.

Dr. Swarnalatha P. completed her M.Tech. Degree in 2006 with first rank in Computer Science and Engineering at VIT University, Vellore. She pursued her doctoral programme in the same institution and obtained her Ph.D. Starting as a Teaching Assistant in 2001. She had a deep involvement in teaching courses and was awarded the best uploading of courses between 2006 to 2009 and had a deep involvement in Image Processing, Artificial Intelligence and Software Engineering and was awarded for the Research Award at VIT University, Vellore. So far she has received 5 awards. Her work on Image Processing for medical and satellite images bought her a funded project on “Development of approaches for geometric, radiometric and atmospheric correction of remote sensing data projects for multi-date data analysis” by Space Application Centre-Ahmedabad at VIT University, Vellore and consultancy work for CMC, Vellore. She was invited by many conferences and for guest lectures for Image Processing, Artificial Intelligence and Software Engineering. She immersed herself in conducting research in chosen area of specialization, guiding Ph.D. projects and M.Tech. students and teaching at undergraduate and post graduate level. Over 70 B.Tech students, 30M.Tech students, and 100 MS students stand testimony for her productivity in Image Processing, Artificial Intelligence and Software Engineering research. She published more than 75+ papers in national and international journals and conferences. She is a Member of many Professional Societies such as Life Member of Computer society of India (CSI) Professional Member of Association of Computing Machinery (ACM), Senior Member of IEEE and Member of ACEEE.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)