International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

RESISTING PASSWORD BASED SYSTEMS FROM ONLINE GUESSING ATTACKS

J. Jayavasanthi Mabel[1], Mr. C. Balakrishnan[2]
[1](PG Scholar) Computer Science &Department, S.A Engineering College, Chennai ,Tamil Nadu;
[2](Assistant Professor) Department of PG Studies, S.A Engineering College, Chennai ,Tamil Nadu.
Email: jjvmabel@gmail.com

Abstract

Authentication to users account to access web services online is achieved using passwords. These passwords are prone to guessing attacks namely brute force and dictionary attacks. Password guessing attack is a method of gaining unauthorized access to one's computer system. Online guessing of passwords is commonly observed in web based applications where users login a number of time to access the details. The guessing attacks on passwords over online are widely spread which reduces the convenience to the legitimate users. Different types of Turing tests are used to prevent legitimate users from such attacks with certain inconvenience to the valid users. On the other hand users also generally prefer common and easy passwords which are weak and make online guessing attacks much easier. The password guessing resistant protocol overcomes these online guessing attacks mainly brute force and dictionary attacks. This is achieved by limiting the number of attempts made during login. The goal is to provide convenient and secured login to the legitimate users which is by blocking the IP address from which there are more number of failed login attempts.

***Keywords*- online password guessing attacks, brute force attacks, password dictionary, ATTs, CAPTCHAs.**

I. INTRODUCTION

Passwords have become the dominant means of access control to online services. The use of passwords is a major point of vulnerability in computer security, as passwords are often easy to guess by automated programs running dictionary attacks. Even though they remain the most widely used authentication method despite their well-known security weaknesses. Online password guessing attacks on websites is a top cyber security risk. User authentication is clearly a practical problem. From the perspective of a service provider this problem needs to be solved within real-world constraints such as the available hardware and software infrastructures. From a user's perspective user-friendliness is a key requirement. A password guessing attack is a method of gaining unauthorized access to a computer system by using computers and large word lists to try a large number of likely passwords. An on-line attack is an attack against an authentication protocol where the attacker either assumes the role of a claimant with a genuine verifier or actively alters the authentication channel. The goal of the attack may be to gain authenticated access or learn authentication secrets.

Various Turing tests are used to prevent password guessing attacks. One effective defence against automated online password guessing attacks is to restrict the number of failed trials without ATTs to a very small number (e.g., three), limiting automated programs (or bots) as used by attackers to three free password guesses for a targeted account, even if different machines from a botnet are used. However, this inconveniences the legitimate user who then must answer an ATT on the next login attempt. Several other techniques are deployed in practice, including: allowing login attempts without ATTs from a different machine, when a certain number of failed attempts occur from a given machine; allowing more attempts without ATTs after a timeout period; and time limited account locking. Many existing techniques and proposals involve ATTs, with the underlying assumption that these challenges are sufficiently difficult for bots and easy for most people. However, users increasingly dislike ATTs as these are perceived as an unnecessary step.

Online guessing attacks on password-based systems are inevitable and commonly observed against web applications.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

Although online password guessing attacks have been known since the early days of the Internet, there is little academic literature on prevention techniques. Account locking is a customary mechanism to prevent an adversary from attempting multiple passwords for a particular username. Although locking is generally temporary, the adversary can mount a DoS attack by making enough failed login attempts to lock a particular account. Delaying server response after receiving user credentials, whether the password is correct or incorrect, prevents the adversary from attempting a large number of passwords in a reasonable amount of time for a particular username. However, for adversaries with access to a large number of machines (e.g., botnet), this mechanism is ineffective. Similarly, prevention techniques that rely on requesting the user machine to perform extra nontrivial computation prior to replying to the entered credentials are not effective with such adversaries.

II. Problem Statement

The purpose is to prevent the online guessing attacks namely brute force and dictionary attacks which aim at gaining an unauthorized access to the valid user's data. This occur when an account is attacked repeatedly. This is accomplished by sending possible passwords to an account in a systematic manner. These attacks are initially carried out to gain passwords for an access or modification attack. There are two types of password guessing attacks.

Brute force attack is the method of trying every possible code, combination, or password until you find the correct one. This is sometime time consuming if the password involves some hash method.

Dictionary attack is the method to guess passwords which is achieved using common list of words to identify the user's password. A dictionary attack uses a targeted technique of successively trying all the words in an exhaustive list called a dictionary that is from a pre-arranged list of values. This uses a dictionary of common words to attempt to find the users password. Dictionary attacks can be automated, and several tools exist in the public domain to execute them.

III. Existing System

Two well-known proposals for limiting online guessing attacks using ATTs are Pinkas and Sander (herein denoted PS), and van Oorschot and Stubblebine (herein denoted VS).

The PS proposal reduces the number of ATTs sent to legitimate users, but at some meaningful loss of security; for example, in an example setup PS allows attackers to eliminate 95% of the password space without answering any ATTs. The VS proposal reduces this but at a significant cost to usability; for example, VS may require all users to answer ATTs in certain circumstances.

ATT challenges are used in some login protocols to prevent automated programs from brute force and dictionary attacks. Pinkas and Sander presented a login protocol (PS protocol) based on ATTs to protect against online password guessing attacks. It reduces the number of ATTs that legitimate users must correctly answer so that a user with a valid browser cookie that is indicating that the user has previously logged in successfully will rarely be prompted to answer an ATT. A deterministic function of the entered user credentials is used to decide whether to ask the user an ATT. To improve the security of the PS protocol, van Oorschot and Stubblebine suggested a modified protocol in which ATTs are always required once the number of failed login attempts for a particular username exceeds a threshold; other modifications were introduced to reduce the effects of cookie theft.

PS Protocol

PS protocol referred to as Pinkas and Sander protocol that requires answering an ATT challenge first before entering the {username, password} pair. Failing to answer the ATT correctly prevents the user from proceeding further. This protocol requires the adversary to pass an ATT challenge for each password guessing attempt, in order to gain information about correctness of the guess.

Initialization-Once the user has successfully logged into an account, the server places in the user's computer a cookie that contains an authenticated record of the username and possibly an expiration data.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

Login procedures:

1) The user enters a username and a password. If his computer contains a cookie stored by the login server then the cookie is retrieved by the server.
2) The server checks whether the username is valid and whether the password is correct for this username.
3) If the username/password pair is correct, then
(a) If the cookie is correctly authenticated and has not yet expired, and the user identification record in the cookie agrees with the entered username, then the user is granted access to the server.
(b) Otherwise, the server generates and RTT and sends it to the user. The user is granted access to the server only if he answers the RTT correctly.
4) If the username/password pair is incorrect, then
(a) The user is asked to answer an RTT with
Probability P (0<P<=1). When his answer is received he is
Denied access to the server, regardless of whether it is Correct or not.
(b) With probability 1-P, the user is immediately
Denied access to the server.

VS Protocol

VS protocol referred to as Van Oorschot and Stubblebine protocol proposed modifications to the previous protocol which track failed logins per username to impose ATT challenges after exceeding a configurable threshold of failures. In addition, upon entering correct credentials in the absence of a valid cookie, the user is asked whether the machine in use is trustworthy and if the user uses it regularly. The cookie is stored in the user's machine only if the user responds yes to the question.

Procedures:

1) User logins with username and password.
2) If there is a cookie in user device then server
Retrieves it.
3) If username/password is correct then
if the cookie is present and valid then login
Successfully
4) else
if Owner Mode(username) or Failed Logins(username)
1 ≥ threshold value then
Send a Turing test to user, login successfully if answer correctly
5) Else login successfully
6) Else
7) Set decision function to TRUE with probability P
8) If decision function or Failed Login(username) 2 ≥ threshold value then
9) Ask a Turing test, wait for answer. Say login
Fails.
10) Else say login fails immediately.

These protocols involve large number of Turing test which an valid user also must undergo which reduces the convenience of the user.

SI as primary units. English units may be used as secondary units (in parentheses). Use a zero before decimal points: "0.25", not ".25". Use "cm^3", not "cc".

Drawbacks Of Existing System

- Attribute Turing tests are generated for each and every login failure.
- Reduce in usability that is user inconvenience.
- The users are traced using the cookies, a name value pair which is a temporary one generated for each and every session.
- If there is more number of failed attempts then it will lead to account locking of the user.

IV. Proposed System

Our main security goal is to restrict an attacker who is in control of a large botnet from launching online single account or multi-account password dictionary attacks. In terms of usability, we want to reduce the number of ATTs sent to legitimate users as much as possible.

The proposal called Password Guessing Resistant Protocol (PGRP), significantly improves the security-usability trade-off, and can be more generally deployed beyond browser based authentication. PGRP builds on these two previous proposals. In particular, to limit attackers in control of a large botnet, PGRP enforces ATTs after a few failed login attempts are made from unknown machines. On the other hand, PGRP allows a high number of failed attempts from known machines without answering any ATTs. We define known machines as those from which a successful login has occurred within a fixed period of time. These are identified by their IP addresses saved on the login server as a white-list, or cookies stored on client machines. A white-listed IP address and/or client cookie expires after a certain time.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

PGRP include the following:

1) The login protocol should make brute-force and dictionary attacks ineffective even for adversaries with access to large botnets (i.e., capable of launching the attack from many remote hosts).
2) The protocol should not have any significant impact on usability (user convenience).
3) The protocol should be easy to deploy and scalable.

PGRP keeps track of user machines by the source IP address. Browser cookies are used in previous protocols to trace the user. Typically; there are drawbacks if no cookie is sent by the user browser to the login server, the server sends a cookie to the browser after a successful login to identify the user on the next login attempt. However, if the user uses multiple browsers or more than one OS on the same machine, the login server will be unable to identify the user in all cases. Cookies may also be deleted by users, or automatically as enabled by the private browsing mode of most modern browsers. Moreover, cookie theft (e.g., through session hijacking) might enable an adversary to impersonate a user who has been successfully authenticated in the past. In addition, using cookies requires a browser interface.

This proposed system is been explained taking online banking system as an example in which the users login a number of time to access their account. The system implementation is been given below.

Most systems implement security in some form or another to preserve privileges for certain users. Authentication of a privileged user without a personal identification scheme that cannot be repudiated is the current mechanism for all but the most secure sites on the Web. We can open accounts on any number of email services, portals, newspapers, and message boards without providing any credentials of our own, such as a passport, driver's license or serial number. In these situations, the first priority may be to point users to the resources they may access; security itself may not take precedence until exploitable details such as credit card information is stored on a given site.

4.1. Tracking Hacker

When there is more of failed login attempts for a particular account than that user is been traced using the IP address. This method find the user's IP instead of the user browser's cookie since cookie can be easily modified and deleted.

The use of IP address is also a tedious process when the request if from a large botnet. Since it involves the process of network address translation. The hacker must be traced carefully when requesting for the resources in the network.

4.2. Generate CAPTCHA

CAPTCHA is the completely Automated Public Turing Test to tell Computers and Humans Apart. When the number of attempts made to login increases beyond three limits a CAPTCHA will be generated. The user must undergo this ATT challenge. This is used as a validation method to verify whether the user is a valid user based on the time taken to complete the challenge. The generated CAPTCHA will be dynamic (i.e.,) new CAPTCHA will be generated for each transaction performed by the user. In this protocol the CAPTCHA generated are the ATTs which will be generated when the user has failed 3 login attempts. This provides a convenient method for the valid user.

4.3. Forwarding New Password

This performs the password generation, which generates new passwords for each transaction so that the account password cannot be traced out by anyone (i.e.,) unauthorized users. This operation is performed after the verification of the user (i.e.,) after the user undergoes the ATT challenge. If the verification is success the generator will generate and forward the new password to the valid user.

4.4. Blocking IP

The users are traced using the IP addresses which are been assigned to the system. If the user's attempt made to login fails even after the new password which is generated then that particular IP address which attempts more failed attempts will be traced and blocked for that particular username. The blocking of IP address is based on time out scheme which makes it convenient to the legitimate users and stop the hackers from guessing the passwords of the user. This makes the user's password more secured from the unauthorized users access.

The system architecture depicts that the user must undergo an ATT only after a limited number of failed attempts made to the login. A captcha will be generated after a three failed login attempts.When the user enters the captcha, the server will collect the details of the particular user and will validate it.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)
International Conference on Information Systems and Computing (ICISC-2013), INDIA.

Once the captcha has been entered a new password will be generated which will be forwarded to the valid users mobile. The password generated will be dynamic for each time it's been generated. If the number of failed login attempts made is more the particular IP will be traced and blocked for that particular user name pair.

This **Fig 1.** represents the entire system architecture:

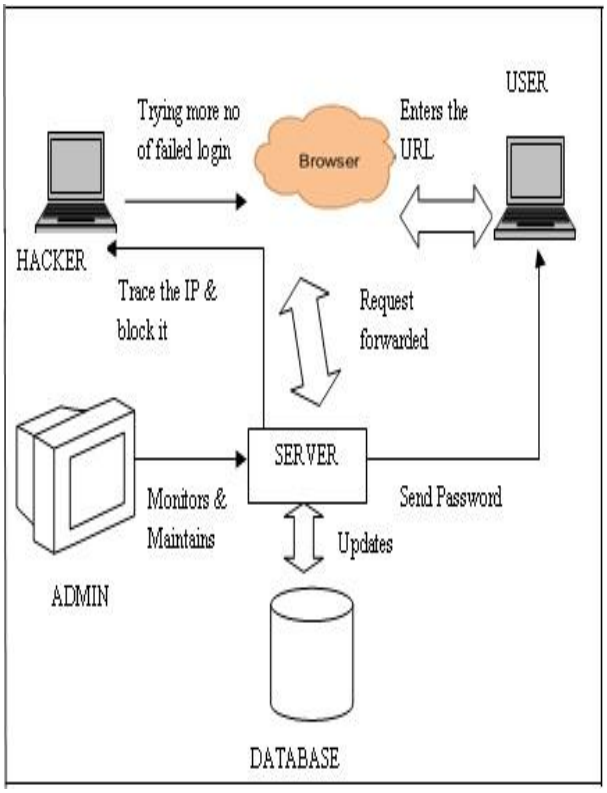

Fig 1. System Architecture

The functional requirements of the system is to resist the online guessing attacks over the passwords which are been achieved using the password guessing resistant protocol. The requirements are to enter the user name and password for checking authorized user or not. If the user name is correct then the User will be successfully logged in. The Server monitors all details during the communication. If the User misbehaves any Login attempt it will be identified and the misbehaved user will be blocked in the network.

Every user are monitored by the protocol so message transmission will be very clear and very interactive to the Server. If misbehave occur from any user, Server will identify the Misbehaving User or malicious login attempt and avoid that user from the communication progress.

V. Conclusion And Future Enhancement

In previous ATT-based login protocols, there exists a security-usability trade-off with respect to the number of free failed login attempts versus user login convenience. In contrast, PGRP is more restrictive against brute force and dictionary attacks. PGRP is apparently more effective in preventing password guessing attacks without answering ATT challenges it also offers more convenient login experience, e.g., fewer ATT challenges for legitimate users even if no cookies are available. This also provides a secured login to the valid users by generating new passwords and forwarding it to their mobile phones. The time taken for generating finishing the ATT challenge is used to verify the authenticity of the user. Blocking IP is an added advantage which is used to overcome the account locking system.

The further enhancement can be done by encrypting the password which is been generated and forwarded to the valid user. Even the encrypted password can be a onetime password which is been generated by the server. This method will be more authenticated which may avoid the password modification or the theft when it is been send from the browser to the valid user.

Acknowledgements

I thank my project coordinator Mr. Muthukumarasamy M.E, my project guide Mr. C.Balakrishnan M.E (PhD), who are members of faculty with the Department of Computer Science and Engineering, S.A Engineering College, without whose guidance, this paper would not have been possible. I also wish to record my thanks to our Head of the Department Mrs.Umarani Srikanth M.E (PhD) for her consistent encouragement and ideas. I would like to express my gratitude to all those who helped me make this paper a reality and gave me the opportunity to publish this paper.

REFERENCES

1. Journal Papers

[1] Y. He and Z. Han, "User Authentication with Provable Security against Online Dictionary Attacks," J. Networks, vol. 4, no. 3, May 2009.

[2] N. Bohm, I. Brown, B. Gladman, Electronic Commerce: Who Carries the Risk of Fraud? 2000 (3) The Journal of Information, Law and Technology.

[3] Chippy.T, R.Nagendran," Defenses Against Large Scale Online Password Guessing Attacks By Using Persuasive Click Points" International Journal of Communications and Engineering Volume 03– No.3, Issue: 01 March2012.

[4] Mathieu Baudet , Bogdan Warinschi , Martín Abadi, *Guessing attacks* and the computational soundness of static equivalence, *Journal* of Computer Security.

[5] International *Journal* of Network Security, Vol.8, Authentication Against *Guessing Attacks* in Ad. Hoc Networks.

2. Text Book

[1] Hacking Exposed: Network Security Secrets &Solutions, 5th Edition by Stuart McClure, Joel Scambray and George Kurtz.

[2] Communication Networks by S.Hekmat.

[3] Improving Web Application Security: Threats and Countermeasures,Mark Curphey.

3. Conference Proceedings

[1] C. Namprempre and M.N. Dailey, "Mitigating Dic tionary Attacks with Text-Graphics Character Captchas," IEICE Trans. Fundamentals of Electronics, Comm. and Computer Sciences.

[2] A. Narayanan and V. Shmatikov, "Fast Dictionary Attacks on Human-Memorable Passwords Using Time-Space Tradeoff," Proc.ACM Computer and Comm. Security (CCS '05).

[3] B. Pinkas and T. Sander, "Securing Passwords against Dictionary Attacks," Proc. ACM Conf. Computer and Comm. Security (CCS '02).

[4] P.C. van Oorschot and S. Stubblebine, "On Countering Online Dictionary Attacks with Login Histories and Humans-in-the-Loop," ACM Trans. Information and System Security.

[5] L. von Ahn, M. Blum, N. Hopper, and J. Langford, "CAPTCHA:Using Hard AI Problems for Security," Proc. Eurocrypt,.

[6] J. Yan and A.S.E. Ahmad, "Usability of CAPTCHAs or Usability Issues in CAPTCHA Design," Proc. Symp. Usable Privacy and Security (SOUPS '08).

[7] L von Ahn, M Blum and J Langford. "Telling Humans and Computer Apart Automatically", CACM, V47, No2, 2004.

[8] T Converse, "CAPTCHA generation as a web service", Proc. of Second Int'l Workshop on Human Interactive Proofs (HIP'05), ed. by HS Baird and DP Lopresti, Springer-Verlag. LNCS 3517, Bethlehem, PA, USA, 2005.

[9] J Yan and A S El Ahmad. "Breaking Visual CAPTCHAs with Naïve Pattern Recognition Algorithms", in *Proc.* Of the 23rd Annual Computer Security Applications Conference (ACSAC'07). FL, USA, Dec 2007.

[10] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber, "How Dynamic Are IP Addresses?," SIGCOMM Computer Comm. Rev., vol. 37.

[11] S. Byers, A. Rubin, and D. Kormann. Defending against an internet-based attack on the physical world. ACM Transactions on Internet Technology, August 2004.

[12] W. Ford and B. Kaliski. Server-assisted generation of a strong secret from a password. In Proceedings of the 9th IEEE International Workshop on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE 2000), pages 176–180, Los Alamitos, CA, USA, June 2000. IEEE Computer Society.

[13] L. Gong. Verifiable-text attacks in cryptographic protocols. In Proceedings of INFOCOM'90, Los Alamitos, CA, June 1990. IEEE Computer Society.

[14] L. Gong, T. Lomas, R. Needham, and J. Saltzer. Protecting poorly chosen secrets from guessing attacks. IEEE Journal on Selected Areas in Communications, 1993.

[15] M. Naor. Verification of a human in the loop or identification via the Turing test. Unpublished manuscript, 1997.

4. Generic Website

[1] Inaccessibility to Captcha http://www.w3.org/TR/turingtest/ (Accessed date: 10-Aug-2012)

[2] Brute force attack http://www.mandylionlabs.com/ PRCCalc/ BruteForceCalc.htm (Accessed date:28-Aug-2012)

[3] Dictionary attacks http://www.cryptosmith.com/node/231 (Accessed date: 02-sep-2012)

[4] Science of Guessing passwords http://www.lightbluetouchpaper.org/2012/05/24/the-science-of-password-guessing/ (Accessed date: 15-sep-2012)