RESEARCH ARTICLE OPEN ACCESS

Password Guessing Resistant Protocol

Arya Kumar[#1], A. K. Gupta[*2]
[#] Student, M.E. Computer, JSCOE, Pune, University of Pune
[*]Associate Professor, JSCOE, Pune, University of Pune

ABSTRACT
Attacks on passwords are increasing day by day. Brute force attack and dictionary attacks are the well known attacks. Automated Turing Test is effective approach to minimize such attacks and identify malicious logins. But sometimes it may create inconvenience to the authorized user as the user always has to cross or go through the ATTs. So to avoid such inconvenience, a new technique called Password Guessing Resistant Protocol (PGRP) is introduced. It overcomes the drawbacks of existing protocols like Pinkas and Sander. PGRP limit the total number of login attempts from unknown source IP address as low as three attempts and the user can make five failed login from the known and frequently used system. CAPTCHA, used is text based, logical and can also be image based. This could make the password guessing more difficult by the automated programs. Multiple ATTs are used to increase the security.

Keywords: online attack, dictionary attack, brute force attack, ATTs, online password guessing attack.

I. INTRODUCTION

Online password guessing attacks are common against web applications. The brute force and dictionary attacks are commonly observed attacks in web applications. In these kinds of attack, attackers run automated password guessing programs. For web login servers, an attacker generally does not have an online attack on a particular account. That is, if the attacker wishes to gain access to the account user ID at the login server he must attempt login. Brute-force attacks are very easily detected. For instance, many web sites institute a three strikes rule where three unsuccessful login attempts will cause access to the account to be locked (at least for some period of time). More sophisticated rules can be applied to detect less obvious attacks, e.g. if the ratio of unsuccessful to successful login attempts exceeds a threshold then particular action to be set up and so on. The brute force attack and dictionary attacks can be avoided by implementing a locking mechanism in the system if it exceeds a number of failed login attempts. But attackers can try only limited number of guesses from a single machine before being locked out, delayed, or challenged to answer Automated Turing Tests (ATTs e.g., CAPTCHAs). ATT is one of the effective defense against automated online password guessing attacks. It restricts the number of failed trials without ATTs to a very small number (e.g. three); this limits automated programs that are used by attackers to three free password guesses for a targeted account [2]. However, this is an inconvenient approach to the legitimate user who then has to answer an ATT on the next login attempt. The users generally feel that being an authorized user and putting correct username and password also he/she has to go through the ATTs. 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 professed as an unnecessary step. Therefore, it has become necessary to implement some method and focus on reducing user annoyance by challenging users with fewer ATTs, while at the same time subjecting bot logins to more ATTs, to drive up the economic cost to attackers. So, a new protocol has been introduced i.e. Password Guessing Resistant Protocol (PGRP). In PGRP, it limits the legitimate users from a known machine to go through a ATTs while enforces ATTs after some failed login attempts. It helps to avoid the bots by enforcing ATTs to users who try to log in from the unknown users and make multiple failed login attempts. This would be more convenient to the authorized users.

II. LITERATURE SURVEY

The literature survey related to this project is given in this section. Turkers first survey asking the some information such as age, native language, education, country of birth, country of residence, years using the internet, frequency of internet use etc. Based on this information, Turkers developed Captchas. They looked at usability issues in presenting audio Captchas to humans. They have a made studies on how much trouble Captchas present for human beings [3].Martin Casado and Michael J.

Freedman have developed and implemented a methodology by which a server can make a more informed decision on whether to rely on IP addresses for client identification or to use more heavyweight forms of client authentication. The main goal is to help a server establish if the IP address of an incoming client is a useful identifier for access-control decisions [5]. Dinei Florencio, Cormac Herley and Baris Coskun examined the question of attacks on password protected web accounts. They concluded that forcing users to choose strong passwords appears misguided and a waste of effort: this offers no defense against the common password stealing attacks. They showed that it is the combined size of the user ID plus password key-space rather than the password key-space alone that protects large institutions against bulk guessing attacks. Greater security for the institution can be achieved by allowing users to keep relatively short passwords, so long as they choose longer user ID's. This reduces the number of break-ins that an attacker with fixed resources can expect, and reduces the burden on users [6].Benny, Pinkas and Tomas Sander proposed that the systems 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. The user has to clear Reverse Turing Test (RTT) before putting username and password. The server checks whether username and password, if it is correct then checks whether the cookies is authenticated or not. If it is not authenticated, then the user has to pass RTT and if it is authenticated then the user is granted the access [4]. Account locking is a usual mechanism to prevent an adversary from attempting multiple passwords for a particular username. Locking is generally temporary; the adversary can mount a DoS attack by making enough failed login attempts to lock a particular account. Pinkas and Sander presented a login 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 will rarely be prompted to answer an ATT. To improve the security of the PS protocol, 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. For both PS and VS protocols, the decision function requires careful design. [7]

- He and Han pointed out that a poor design may make the login protocol vulnerable to attacks. The authors proposed a secure nondeterministic keyed hash function as decision function so that each username is associated with one key that should be changed whenever the corresponding password is changed.[1]

III. PROPOSED WORK

ATT is one of the effectual defenses against automated online password guessing attacks [7]. However, this creates inconveniences to the legitimate user who then must answer an ATT on the login attempt. To improve this, a new protocol is proposed called Password Guessing Resistant Protocol (PGRP). It significantly improves the security, usability trade-off, and can be more efficient beyond browser-based authentication. To limit attackers in control of a large password attacks PGRP enforces ATTs after a three failed login attempts are made from unknown machines. On the other hand, PGRP allows a five of failed attempts from known machines without challenging to any ATTs. Here known machines are those from which a successful login has occurred within a fixed period of time. The machines are identified by their IP addresses saved on the login server as a white list. A white-listed IP address expires after a certain time span. Tracking users through their IP addresses also allows PGRP to increase the number of ATTs for password guessing attacks and meanwhile to decrease the number of ATTs for legitimate login attempts. The PGRP protocol maintains a data structures containing log information. It maintains a white list of IP addresses and failed login attempts from a known system and failed login attempts from known systems.[4]

3.1 Features

- The proposed protocol minimizes users inconvenience in the login process
- It has a white list of IP addresses.
- If a user sees that someone has tried to log in to his account and have made failed log in attempts then the user can add that IP address to the blacklist. This list may be made only by tracking the log information. This list may consume considerable memory;

3.2 Goal of PGRP

The goal of PGRP is as follows:

1. PGRP protocol should make brute force and dictionary attacks ineffective for opponents.
2. The protocol should not make the legitimate users, any additional steps other than entering login information. The security must be increased with minimal effect in decreasing the login usability.
3. The protocol should be easy to organize and setup and scalable, requiring minimum computational resources in terms of memory, processing time, and disk space etc.

The 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.

1. First Time Login from an Unknown Machine:- If a valid username and password pair is provided from an unknown machine (i.e., a system from which no successful login has occurred within a defined period), no ATTs are required if the total fail count from unknown machines is below within a time limit. This threshold may be exceeded as follows:

(a) The user may provide incorrect passwords from that machine maximum times.
(b) Attackers may have attempted failed passwords the threshold (from unknown machines); or
(c) A combination of a) and b). Once a user successfully logs in, then that machines IP address is added to the white list (W).

2. Subsequent Login from a Known Machine: - ATTs are sent to a known machine only when more failed login attempts are made. For that machine and the user account is possibly under attack. The legitimate users may make a reasonable number of password mistakes without experiencing any ATTs.

3. Valid Password Is Provided:- Users may be annoyed if they provide a valid password, and yet has to answer an ATT. When a valid password is provided by the user, no ATT challenges are sent if the attempt comes from a known machine which has not been used for more than defined failed login attempts within a time period. If the user hits or crosses the threshold, still no ATTs are sent if the number of failed login attempts from unknown machines remains below the threshold. Thus, users must pass ATT challenges only when they attempt login from unknown machines and the number of failed attempts from unknown machines has hit or crossed the threshold.

4. Invalid Password:- This may be a common occurrence for several reasons such as users may need multiple attempts to recall the correct password, users cycle-through multiple passwords due to multi password interference and typing errors including activating the caps lock key. From each known machine, a user is allowed up to some limited number of attempts, before challenged with ATTs.

5. Invalid Username: - When a user tries login with a nonexistent username (e.g., typing errors), an ATT challenge is given, irrespective of the password or ATT answer, the login fails. This feature restricts attackers from learning valid usernames and improves protocol performance in terms of memory usage. This type of error would be limited in practice because usernames, in contrast to passwords, are echoed on a display.

3.3 Data Structures

It maintains three data structures:

1. W: - A White list of source IP address, username pairs such that for each pair, a successful login from the source IP address has been initiated for the username previously.

2. F_U:- Each entry in this table represents the number of failed login attempts for a valid username, un.

3. F_K:- Each entry in this table represents the number of failed login attempts for each pair of (src IP, un). Here, src IP is the IP address for a host in W and un is a valid username attempted from src IP. A maximum failed login attempts are recorded; crossing this threshold may mandate passing an ATT.

Each entry in W, F_U, and F_K has a write-expiry interval such that the entry is deleted when the given period of time (t1, t2, or t3) has lapsed since the last time the entry was inserted or modified. There are different ways to implement write-expiry intervals. A simple approach is to store a timestamp of the insertion time with each entry such that the timestamp is updated whenever the entry is modified. At anytime the entry is accessed, if the delta between the access time and the entry timestamp is greater than the data structure write-expiry interval (i.e., t1, t2, or t3), the entry is deleted.

3.4 Contribution

ATT itself is a very strong approach to avoid brute force attack and dictionary attacks but to make it more powerful the following contribution is added to this project:

1. Multiple ATTs:- If the user has to come across ATTs due to failed login attempts beyond the threshold limit then the user has to go through multiple ATT test. For example, the first ATT would be of normal text based CAPTCHA, if the user clears that then the second ATT would be of some logical answering questions (i.e. 2+3=?, 4*5=? etc) and if the user clears this also then he has to appear for next ATT which would consists of personal questions (i.e. the user has to select the question from the list of questions which he/she has answered during creating an account.)

2. Inaccessible site: - If the user make multiple failed login attempts as well as fail to clear the ATTs then

no user will be allowed to access the site from that IP address. The access to that site would be blocked for some period.

3. User log information is provided:-The server provides each user is provided with the user log information. The user log information consists of the information about users log in date and time with the IP address from which the login attempt was done. It also provides the log status whether the login was successful or whether then attempt had failed. This is really very useful information as the legitimate user could make out whether the failed login attempt was made or not. The server maintains these records of all the users.

3.5 Algorithm

Steps are described here, which gives the idea about the working of the project. Parameters used in the algorithm are as below with its meaning.
t1 = number of login attempts
un= username of the user

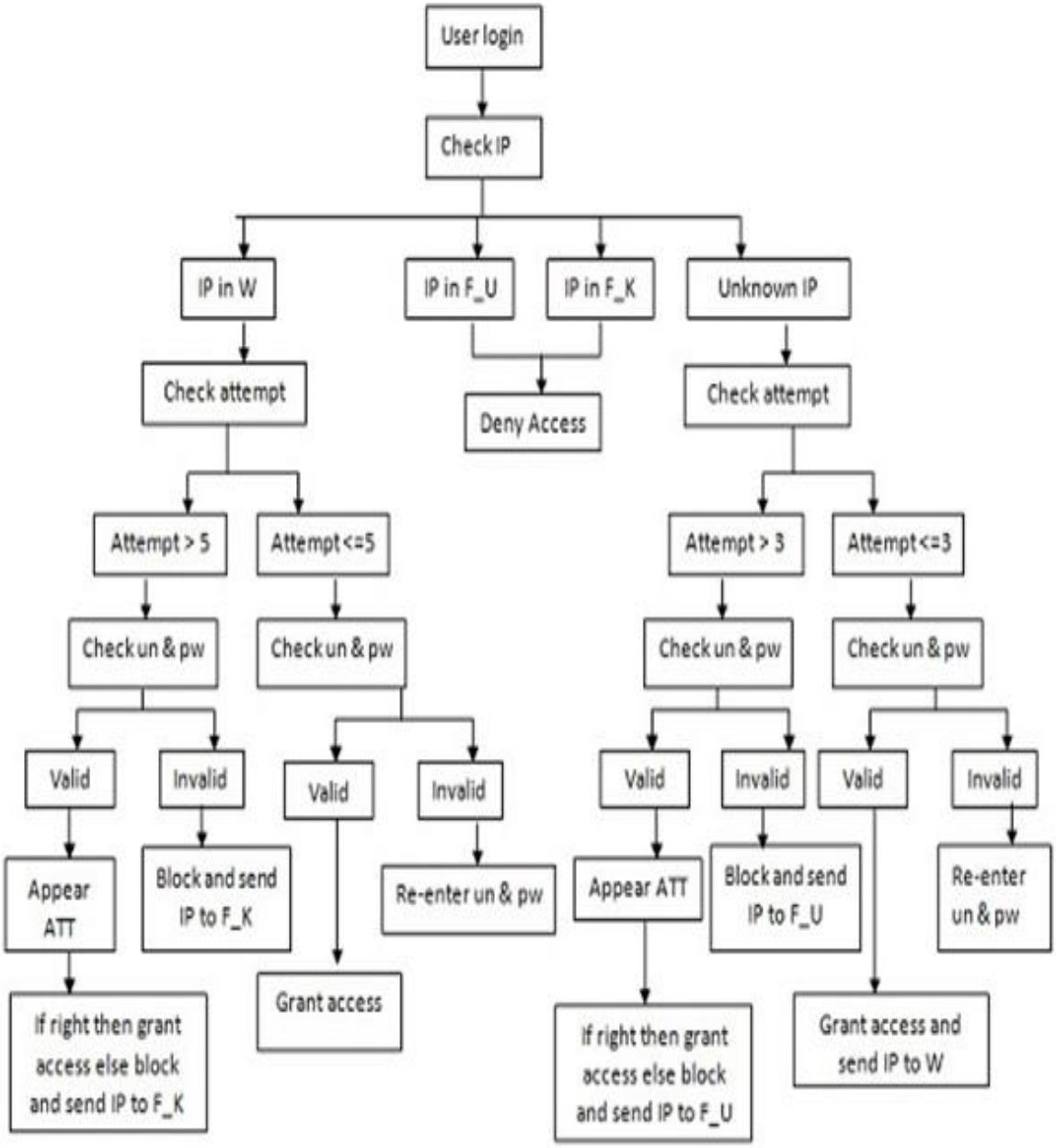

Fig.1. Overall working of the proposed system

pw= password of the user
W= White list data structure
F_K= data structure containing failed login from the known system
F_U= data structure containing failed login from the unknown system

Steps that take place when login is done from a Unknown Machine:

1. Read credential such as un and pw, if the entered username (un) and password (pw) is correct at t1=1, then add the IP address of the system to the W.
2. Then allow the user to access the information which authorized to that particular user.
3. If the entered user name and password is not correct at t1=1 and the IP address is not present in W then the user can make t1 is less than equal to three login attempts. If still the username and password comes out to be invalid then the IP address of the system is sent to F_U and the user has to go through the ATTs.
4. Then again he is given a chance to log in to the account; if the user name and password is correct then the user has to go through the ATTs.
5. If the user fails to attempt the first ATT then the user has to crack the another and if he successfully go through that ATT he has to again go through multiple ATTs like (text based CAPTCHA, logical answering questions, personal answering questions etc.)
6. Then the user is grant access to the system.
7. If the user makes failed login attempts t1 greater three then the IP address is sent to F_U and no user will be allowed to access the site for a period of time from that IP address.

Steps that take place when login is done from a Known Machine:

1. Read credential such as un and pw, if the entered username and password is correct and is provided from a IP address which is present in W and t1 is less than and equal to 5, then the user is grant access as per the privilege provided to him/her.
2. If the entered username and password is incorrect for t1 is less than and equal to 5, then the IP address is deleted from W list and it will be added to the F_K list.
3. The user will be blocked for some period of time.
4. If again a login attempt is made from the same IP then the user has to go through the multiple ATTs described above then if all entries are correct then access is granted.

IV. CONCLUSION AND FUTURE SCOPE

Though ATT based technique is efficient in dealing with brute force and dictionary based attacks PGRP protocol enhances the efficiency of the technique and makes the system more restrictive against the attacks. It helps in a convenient login process for the authorized users as authorized users need not to go through the ATTs from a known machine which increases the usability. Due to the use of multiple ATTs the systems security is increased. The log information provided to the legitimate users helps to give all the log successful as well as unsuccessful log information so that the user can protect his account by changing the password if some illegal access is made. It can be used in small as well as large organizations. In future, SMS alerts can also be provided to increase the security of the system.

V. ACKNOWLEDGMENT

We thank anonymous referees whose comments improved this paper.

REFERENCES

[1] Y. He and Z. Han, "User Authentication with Provable Security against Online Dictionary Attacks", *J.Networks,* vol. 4, no. 3,pp. 200-207/May 2009.
[2] M. Motoyama, K. Levchenko, C. Kanich, D. Mccoy, G.M. Voelker, and S. Savage, "Re: CAPTCHAs Understanding CAPTCHA-Solving Services in an Economic Context", *Proc. USENIX Security Symp.*, Aug. 2010.
[3] E. Bursztein, S. Bethard, J.C. Mitchell, D. Jurafsky, and C. Fabry, "How Good Are Humans at Solving CAPTCHAs? A Large Scale Evaluation", *Proc. IEEE Symp. Security and Privacy* May 2010.
[4] B. Pinkas and T. Sander, "Securing Passwords against Dictionary Attacks", *Proc. ACM Conf. Computer and Comm. Security (CCS 02),* pp. 161-170, Nov. 2002.
[5] Martin Casado and Michael J. Freedman, "Peering Through the Shroud: The Effect of Edge Opacity on IP-Based Client Identification", *Stanford University.*
[6] Dinei Florencio, Cormac Herley and Baris Coskun "Do Strong Web Passwords Accomplish Anything?" *Proc. HotSec 2007.*
[7] Mansour Alsaleh, Mohammad Mannan, and P.C. van Oorschot, Member, IEEE, "Revisiting Defenses against Large-Scale Online Password Guessing Attacks", *Published by the IEEE Computer Society*,January/Febbruary 2012.