ISSN:0975-9646

Secure Password Transfer Using Randomized Grids

Aranta Rokade, Sandesh Gupta, Supriya Rokade, Vidhi Sanghavi

Department of Information Technology, Fr. C.R.I.T.
Vashi, Navi Mumbai, India.

Abstract —**Security is one of the major concerns of man since his existence. Be it for his life, as in the olden days or for his personal information, as in the recent years. But, as security measures are advancing, so are the ways in which data (example passwords, session content, etc.) can be hacked. Thereby, the needs to develop secure ways of information exchange. This paper explains a secure way in which a password can be exchanged between a client and a server without passing it in its plaintext form. In this technique, once a user sets the password, the user never types it again i.e. the actual password is never sent to the server. It travels in a coded form which is different for every login session and is derived from a random grid. This grid is generated at the server by an algorithm and gets refreshed after every login attempt. This makes the current login password independent of the grid. Thus the major advantage is that even if a packet carrying password is sniffed (leaked), the user's original password is never revealed. Also, the length of the password gets reduced without compromising its confidentiality. In spite of this, this technique provides a secure way of password exchange between a client and server.**

Keywords— **security, client, server, login, grid, confidentiality.**

I. Introduction

One of the major concerns of data is keeping it secure. The three pillars of security are Confidentiality, Integrity and Authentication. Information security is must in order to prevent malicious attacks, interruptions, modification etc. It is very essential to authenticate i.e. to check whether the correct user has access to the data. The methods to authenticate are passwords, smart cards and biometrics. Passwords are the most convenient to use. While the other methods require hardware support and so it becomes expensive and identification process can be slow and ineffective.

Passwords too have drawbacks like eves dropping, masquerading, shoulder surfing etc. So to avoid this, passwords can be made strong by making them complex i.e. increasing the length or inserting more characters or symbols. But then it is very difficult to remember such passwords.

II. Password Issues

Although passwords are the most convenient way of authentication they too have certain issues. The various issues faced while dealing with passwords are:

2.1 Too many passwords to remember

This results in password reuse. Thus, cracking one account will result in cracking subsequent accounts.

2.2 Default Passwords

Failure to change default passwords, allows access to unauthenticated users.

2.3 Social engineering

Social engineering is the art of manipulating people so they give up confidential information. The types of information these criminals are seeking can vary, but when individuals are targeted the criminals are usually trying to trick you into giving them your passwords or bank information, or access your computer to secretly install malicious software–that will give them access to your passwords and bank information as well as giving them control over your computer.

2.4 Error logs

Error logs may contain "almost" passwords causing loss of confidentiality of the password.

2.5 Keylogger

A software installed in a computer which taps all the keys strokes on the keyboard and stores information about the keystrokes which can be used to hack personal accounts. So it is always advisable not to access personal accounts in public computers.

2.6 Shoulder Surfing

This form of an attack is initiated by someone over shoulder watching the user enter their password.

2.7 Password crackers

It is process of recovering passwords from data that have been stored in or transmitted by a computer system. Common methods of password cracking are brute force, dictionary attacks, pattern checking, word list substitution, etc.

III. Types of attacks during password transmission

Once a password is correctly entered by the client, there are different attacks during that packet's transmission from the client to the server. These attacks can access the packet's data i.e. the password and can reuse it to access that client's account later. The various attacks possible on the packet are as listed below:

3.1 IP spoofing:

IP spoofing means creation of internet packets with fake IP addresses. Internet Protocol is the basis of internet which contains source IP address and destination IP address. Once the source IP is forged and when the packet reaches the destination there are chances the destination might be hacked.

3.2 Sniffer attacks:

A sniffer is an application or device that can read, monitor the data, capture the data exchanges and can read network packets. That is it allows the full view of the data that resides inside the packets when the data is being send from client to server. It allows the attacker to read the communication or sometimes it may corrupt the network.

3.3 Hijacking (man-in-middle attack)

When the client sends the message to the server, there is someone who actively monitors the activity of the client. If the data which is been send is in plain text format then the attackers can easily re-route the data due to weak network layer and can have a full access for the information. For example, session fixation, session sidejacking, cross-site scripting.

3.4 Trojans

It is ordinary software but performs unintended activities or malicious attacks due to which there is loss of integrity.

IV. Effects of these attacks

- Comprise of confidentiality.
- Loss of data by deletion of information.
- Denial of service attacks.
- Modification in the information.
- Misuse of the data

V. Prevention Of The Attacks

The method to prevent the compromise of passwords during its transmission is to encrypt it and send it from the client to the server. Common methods of encryption have certain problems associated. These problems are discussed below:

5.1 Substitution Cipher:

These are easy to decode using frequency analysis. Vulnerable to frequency attacks and brute force algorithms.

Management of Keys:

5.1.1 Caesar Cipher and Vigenere

Only 25 keys are available and key length is very small.

5.1.2 One time pad:

Key should be as long as the amount of data.

5.1.3 Random Key Generation

Difficult to produce big amounts of truly random keys.

5.1.4 Key exchange

The key must be known by both sender and receiver.

5.2 Transposition Cipher

By performing some sort of permutation on the plaintext letters or by multiple anagramming. Keys close to the actual key will reveal long sections of legible plaintext.

5.2.1 Symmetric keys

Uses the same key at the sender and the receiver, thus leading to non-repudiation.

5.2.2 Asymmetric keys

Leads to information disclosure on sign and encrypt and spoofing threat on encrypt and sign. It also involves added features of key exchange, certificates and signatures.

5.2.3Hashing

This technique is prone to dictionary attack and collisions.

VI. Grid method

This section explains the proposed process, for transferring a password in a secure manner from a client to the server during the registration and the login process.

6.1 *Registration Process Steps*

This process is to be followed when user is registering to any login system for the first time. The form will contain fields such as enter username, password, confirm password, security questions.

6.1.1 User enters text of his choice in the username field in the form given to user.

6.1.2 User has to answer the security questions asked. These questions will increase the security level during the login process.

6.1.3 A randomized grid is generated at server side and is then sent to client machine as shown in Fig 1. This randomized grid contains random characters allotted to random cells.

	0	1	2	3	4	5	6	7	8	9
0	A	R	W	X	I	d	1	V	^	/
1	E	B	:	`	T	j	6	J	C	N
2	9	Q	+	P	~	Z	(	|	]	\
3	G	‘	C	“	_	%	8	!	P	2
4	3	S	T	G	F	]	Y	L	M	O
5	W	L	K	*	N	S	{	[	)	;
6	M	D	4	[	O	Q	H	}	h	X
7	Y	7	I	K	-	z	<	@	?	µ
8	F	J	>	Z	$	=	E	,	k	÷
9	R	&	V	#	U	5	.	b	≠	∑

Fig 1. Randomized Grid for Registration (row and column numbers not visible to the user)

6.1.4 The user has to select the characters of his new password from the grid given.

6.1.5 Client Side Computation

These steps are performed by the client machine after the above steps are completed:

6.1.5.1 The row number and column number is found out for each character of the password.

6.1.5.2 The row and column numbers for every character in the password are then combined to form the session password during the registration process.

Example:
Password is: A%z
Row and column of ‘A’ = 00
Row and column of ‘%’ = 35
Row and column of ‘z’ = 75
The session password is: 003575

6.1.5.3 The session password is then sent to server.

6.1.6 Server Side Computation

The server maintains the same instance of the randomized grid. The following steps are performed by the server after it receives the session password from the client:

6.1.6.1 The characters of the session password are grouped into pairs.

6.1.6.2 Of every pair formed, the first character denotes the row number and the latter denotes the column number.

Example:
Session password is: 003575
Grouping done as: <u>00</u> <u>35</u> <u>75</u>

6.1.6.3 Thus, for every pair in the session password, the actual character is calculated and the original password is generated.

Example:
00 => A; 35=> %; 75=>z
Thus, the original password is: A%z

6.1.6.4 The original password and the answers to the security questions are then stored in the database.

6.2 *Login Process Steps*

This process is followed to login to the session of the website.

6.2.1 User enters the username and answers the security questions in the login form. These values are then sent to server for verification.

6.2.2 At server side, database compares the username and security answers.

6.2.3 If the values match, then randomized grid is generated at the server side and sent to the client, as shown in Fig 2. If the values don’t match, then the access is denied.

This randomized grid contains random characters and random place values allotted to random cells. The place values are <u>NOT</u> visible to the user.

a ^{13}	d ^{45}	J ^{66}	M ^{33}	K ^{42}
b ^{32}	S ^{37}	x ^{21}	> ^{27}	O ^{61}
P ^{55}	W ^{12}	] ^{76}	L ^{81}	\ ^{16}
G ^{9}	E ^{8}	R ^{4}	K ^{2}	. ^{43}
T ^{34}	F ^{46}	{ ^{69}	& ^{78}	@ ^{20}
; ^{11}	Q ^{70}	A ^{80}	B ^{88}	I ^{60}

Fig 2.Randomized Grid for Login (place values will not be visible to the user)

6.2.4 Upon getting the randomized grid from the server the user selects the password characters from it.

6.2.5 Client Side Computation

6.2.5.1 The password entered by the client is grouped into pairs. If there is odd number of characters in the password then an additional filler character is added in the end. This filler character can be a character chosen dynamically during the grid generation or can it be a static selection.

Example:
Password entered by user: WOdF@
After grouping into pairs: <u>WO</u> <u>dF</u> <u>@X</u> …..X->filler

6.2.5.2 For every pair of the characters of the password entered by the user, the intersection is determined and a new character for every pair is formed. The intersection is intersection of the row of the first character in the pair and the column of the latter character in the pair.

Example:
WO=>\ dF=>d @X=>{

6.2.5.3 A place value is assigned to every cell and the sum of the place values of all the characters of the password entered by the user is calculated.

Example:
Sum of place values of WOdF@ = 12+61+45+46+20 = 184

6.2.5.4 The sum of the place values is appended at the end of intersections, which forms the session password. This session password is then sent to the server side.

Example:
Thus, the session password based upon user input is: \d{184

6.2.6 Server Side Computation

The sever retrieves the actual password of the user from the database and carries out the following process, after the server receives the current session password from the client.

6.2.6.1 In backend, the same procedure is performed on the actual password as mentioned in section 6.2.5

6.2.6.2 The coded password generated at the server is compared with the one received from the client. If both the encrypted passwords match, then the access is granted, else the access is denied.

VII. ALGORITHM

The algorithm for forming the grid is:

1. Declare an array pass_chars[N] which stores all the available characters, numbers and symbols used for a password.

```
char pass_chars[N];
pass_chars[]={'a','b',...'A','B'...'1','2',..'@','#'...};
```

2. Declare an array pass_values[N] which stores all the possible place values used for every cell of the grid.

```
char pass_values[N];
pass_values[]={1,2,3,4,...100,101,102,....};
```

3. Declare an matrix of m x n size which will be the randomized grid given to the user.
Where, n*m=N.

```
char grid[n][m];
```

4. Assign random characters and place values to all the cells in the grid.

```
for(i=1;i<=m;i++)
   for(j=1;j<=n;j++)
   {
      grid[i][j] = random(pass_chars[]);
      //generate random characters from pass_chars[]
      delete grid[i][j] from pass_chars[];
      grid[i][j] = random(place_values[]);
      /*generate random place values from
        place_values[]*/
      delete grid[i][j] from place_values[];
}}
```

5. Function: char random(array[]) generates a random element from the array[] passed to it as a parameter.

VIII. POSSIBILITIES OF GETTING RANDOM GRIDS

If we have a grid of n x m size, then the no. of grids possible are:

$$=(n \times m)!$$

For example, if there are 7 rows and 10 columns, then the no. of grids possible are:

$$= (7 \times 10)!$$
$$=70!$$
$$= 1.1978571669969891796072783721689 \times 10^{100}$$

IX. POSSIBILITIES OF COLLISIONS

For every character as an intersection of row and column, the numbers of possible combinations resulting in the same intersection are:

$$= {}^{N}C_{1} \text{ X } {}^{M}C_{1}$$
$$= N \text{ X } M$$

Eg. If there are 7 rows and 10 columns, the numbers of possible combinations resulting in same intersections are:

$$i.e.\ {}^{10}C_{1} \text{ X } {}^{7}C_{1}$$
$$=7 \text{ X } 10$$
$$=70$$

Thus, to make sure that the session password's intersection is obtained from the original password and not any other combination, the place values are used.

X. ADVANTAGES

The advantages of this mechanism for password transfer between client and server are:

- A session password is transferred for every login attempt thus avoiding the transfer of the original password.
- Thus, an encrypted password is sent which is valid only for 1 login attempt.
- Avoids the compromise of the passwords by various attacks such as :
  1. Keylogging
  2. IP - spoofing
  3. Sniffer attack
  4. Session hijacking

Thus, even if the password for the current session gets hacked, the original password is still safe. This is because the session password can’t be cracked to get the original password. Thereby, avoiding further compromise of the security of the user account.

- Reduces the issue of collisions by the use of the place values appended. This confirms that the session password is generated from actual password only and not by any other combination.
- No management of keys is required.
- No external hardware required as in the case of biometrics.
- No additional costs required.

XI. Conclusions

This paper presents a technique which can securely transfer passwords from client to the server. The major advantage of this method is that even if a packet carrying password is sniffed (leaked), the user’s original password is never revealed. It avoids various attacks such as packet snooping and hijacking. Also the length of the password gets reduced without compromising its confidentiality. This method can be implemented for any login website. Thus technique provides a secure way of password exchange between a client and server.

Acknowledgments

We wish to express our sincere gratitude to teachers, friends and member of organization for their kind co-operation and encouragement.

References

[1] Mark Stamp’s Information Security by Deven N .Shah

[2] www.wikipedia.org

[3] Communication and Network Security by Sumeet Kasera.

[4] email_hackingwekipedia

[5] secret drawing encyclopedia.

[6] Anatomy of hacking-website.

[7] CONSUMER SURVEY: PASSWORD HABITS :A study of password habits among American consumers . September 2012 .www.csid.com