Enhanced Authentication Scheme using Password Integrated Challenge Response Protocol

Nitesh Rastogi
(Assistant Professor)

Avinav Pathak
(Assistant Professor)

Shweta Rastogi
(Assistant Professor)

IIMT Institute of Engineering and Technology, Meerut

ABSTRACT

Authentication is a process by which both the sender and receiver check and identify the validcommunicative partners prior to initialization of message exchange. Authentication is a security module which is defined at the time of starting of communication between the two communicating entities such as client and server due to the unlimited amount of insecure and malicious intruders it becomes significant to protect the network communication. The protocols are used here for initializing the valid keys and passwords for communicating with the devices. Here we are proposing a new and more secured authentication scheme based on unique identification of every pair of client and server which have entered communication on a global network. This method enhances the present security scenario by diminishing the replay, modification attacks as well as server eavesdropping. This method also enhances the efficiency, integrity, reliability and security in authentication process.

Keywords:

Challenge Response, Authentication, Password, Public/Private Keys, Message Digest, Unique List.

1. INTRODUCTION:

In today’s world there are a vast variety of computer networks by which multiple number of users share and communicate their specific information types. There is always continues need of protection for such transactions. Multiple security protocols and schemes have been implemented to accomplish such security demands. A Challenge Response protocol is a scheme that provides security against passive eavesdropping. There is a provision that passive intruder is unable to discover the correct password during communication. In this protocol, there are two users A and B that are trying to authenticate each other. To initiate this protocol there is a message exchange amongst the entities (A and B). Entity B sends a challenge to A. Since B is using a separate challenge for each executable protocol, there is a less frequent chance of eavesdropping.

In order to prevent the dictionary attacks, the scheme is using encrypted key exchange mechanisms. A public/private key pair is generated and communicated across the channel only at a single moment of communication initialization between the client and server. Each key pair is unique and randomly generated that minimizes masquerading of users. The passwords are also verified as per the unique identification list generated for pair of user. After that an integration of both the mention schemes is passed between the users A and B via encrypted information function that is message digest. On receiving the challenge and the password, the server/user is now satisfied with the user authentication identity and accepts the communication password for further message transaction. Each time a new user entered the communication unique list of keys and password is generated to establish communication.

Section 2 describes the overall work initiated in this scheme by spheres of researchers. It describes the instances of work undertaken and completed. Section 3 describes the proposed work on the topic. It defines the algorithms and working of the new work implemented in this research scheme. The explanation to the algorithms and the methods employed are described in detail. The figures are used to give clarity. Section 4 and 5 discuss the conclusion and the future scope of the paper and the scheme.

2. RELATED WORK:

A large number of Cryptographic protocols[1] depend on passwords which are selected by the people for better authentication. There is an observation that multiple users gain access over the internet in order to attain required use full information which can be considered volatile for the outside word. Multiple malicious intruders[2] are present to disrupt the communication networks. The password base protocol problem was first studies by Gong et al. Another very important work was done by Bellovin and Merritt[3] in encrypted key exchange (EKE) in the year 1993. The method uses password exchange via challenge response protocol. This scheme uses challenge response mechanism in order to identify the legal user. The basic authentic scheme was exposed to multiple security threats. Later on Bellare and Rogaway[5] suggested several new parts of the Ideal Cipher in their proposal to IEEE P1363.2.Since then Better authentication scheme were proposed by Jablon 1996, Steiner et al. 1995 and Patel 1997. This scheme for security of these solutions was based on assumptions. Other methods that were developed include session key exchange[6] methods for better and secured authentication using challenge response protocols version. The scheme formulated peer authentication process. The next versions were not more practically feasible in communication networks.

3. PROPOSED WORK:

This paper proposes a dynamic authentic scheme for network communications. This authentication scheme introduces a method of password based authentication of user’s messages and the implementation of challenge response protocol makes the scheme less vulnerable to the security threats. Here in this scheme the key exchange is done by using message digest for key transfer this method uses better encryption of key pairs and eliminates the chance of replay and masquerade attacks. The password is exchanged after wards and a client is given permission to transfer messages which is better and prioritized secured method.

The present paper preludes with the complete integrated mechanism of massage transfer via authentication of password. The procedure form the beginning of message transfer has been defined as:

3.1 Key Generation Scheme:

In order to authenticate the password transaction secured key exchange and generation is the first priority to discourage intruders attack. The scheme can be understood as:

Step 1: An initial input key is taken as 64- bit key. The output key that we gain after the process is of 32-bits.

Step 2: The key pair is divided into two halves and the right shift of the bits is done so that the first 4 -bits are initially shifted.

Step 3: A parity drop is initialized that reduces the no. of bits.

Step 4: Again removal of parity bit results in reduction of the generated key. After sequential key reduction the bits are 32 as output key.This key unique and more secure for data transaction.

Algorithm:

Parameters: (LP,RP)=left and right halves of key, K=initial key, K'=key after first parity drop, K''=key after second parity drop, P=Parity Drop, P' =new Parity drop, Fk=final key

K_GEN()

```
{
Input= 64bit key;
output=32bit key;
Begin
{
Divide the input key (K) in two halves;
LP=32;
RP=32;
(LP,RP)>>4// Perform the right shift on the key
Divide this key obtained;
K'=P;          //Perform paritydrop (P) on the key (i.e.64-bits)
Key=56-bits;                    // New 56-bit key
Eliminate the parity bit from the key;
(LP,RP)<<4     //Perform again left shift on the key
K'=P'        //Perform new parity drop on the 56-bit key
K''=48-bits;                    // New 48-bit key
K'''=32-bits;                  //Perform S-box permutation
Final key obtained=Fk=K''';    //   Length (Fk)  is 32-bits
End ();
        }}
```

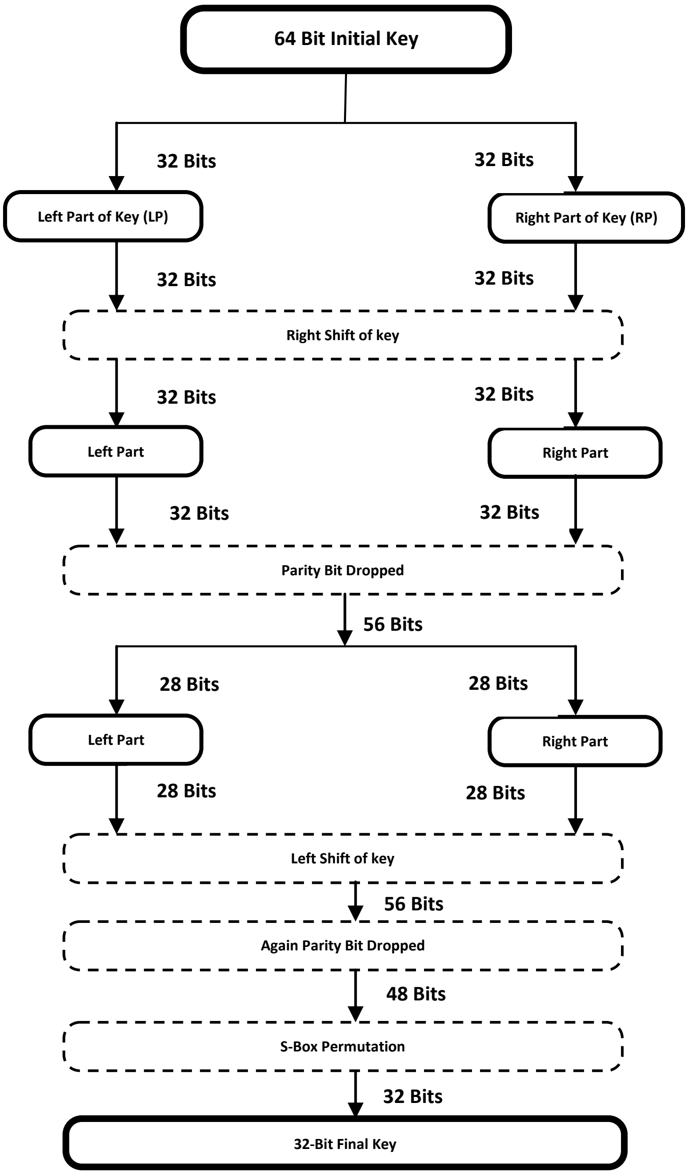

Figure 1.0 Key Generation

3.2 Unique Password Generation:

This instance derives a unique list of user and server passwords so that we can provide better communication without any malicious affinity. The stepwise explanation to the process is:

Step 1: On the onset, it allocates a new set of passwords every time a new user registers with the server during the communication or message transaction.

Step 2: The password is generated as a combination of alphanumeric sequences dependant on the user demand. Each new user can derive the new set of passwords by allocating the string pair and then transferring this generated password to the server it wants to communicate as demonstrated on the algorithm of the phase.

Step 3:When the session server is authenticated with the new user, it is allowed to transfer the information among themselves.

Step 4: Every time a new user wants to initialize its communication, it exchanges encrypted passwords within themselves and then a response is allocated to such an authenticated client. Next, the data exchange starts among the user and the server that is authenticated with the tunnel.

Step 5: After communication termination, the passwords are destroyed and a new password is again allotted to new client/server pair. This maintains the uniqueness of the information exchange.

Parameters: N[]=input text, p=new password string list, p'=final password, str1, str2. str3=strings of password

Input: 32-bit text

Output: Newly generated password of 32-bits

```
P_GEN();
{
int N[32],i;
char str1[], str2[],str3[],p,p';
for(i=0;i<=32;i++)
{
If(N[0]=='/0')
    {
terminate the process;
                        //check for null value
    } else
{
N[0]=Numeric value;
}
}
gets(str1); // input string of password
strlen(str1);
strlen(str2);
if(strlen(str1)<33)             //check out of bounds
{
strcat(str1,str2);
}else
{
terminate;
}
Generate the password=str2;
}
gets(str3);
strlen(str3);
if(strlen(str3)<33)
strcat(str3,p);
}else
{
terminate the process;
}
Final password=p;
}
Select form the list of p= 1 to 32;
//New password list
printf("Unique password",p');
 //output password
return();
}
```

3.3 Exclusive OR Scheme:

In This scheme, we use XOR function with generated 32-bit final key (Fk) and Unique Password (P') then the result of XORed function is new generated Key (FKP'). Which will we use in authentication for user.

Principles of XOR function:

In cryptography, the **simple XOR cipher** is a type of *additive cipher*, an encryption algorithm that operates according to the principles:

$A \oplus 0 = A,$

$A \oplus A = 0,$

$(A \oplus B) \oplus C = A \oplus (B \oplus C),$

$(B \oplus A) \oplus A = B \oplus 0 = B,$

Where $\oplus$ denotes the exclusive disjunction (XOR) operation. This operation is sometimes called modulus 2 addition (or subtraction, which is identical). With this logic, a string of text can be encrypted by applying the bitwise XOR operator to every character using a given key. To decrypt the output, merely reapplying the XOR function with the key will remove the cipher.

3.4 Encryption Scheme:

This scheme employees the secured transfer of both the password and the XORed key (Fkp') with the message to the selected or specified user. The following steps are undertaken in this scheme:

Algorithm –

1. X: (Ex(Fk), Dx(Fk)).
2. X→Y: X, P' (Ex(Fk)).
3. Y: Compute Ex(Fk) = P' (Ex(Fk)) /P' ; Generate random secret key FkP'xy
4. Y→X: P' (Ex(Fk)(FkP'xy)).
5. X: FkP'xy = Dx(Fk) (P' (Ex(Fk) (FkP'xy)))/ P'; Generate unique challenge Cx.

6. X→Y: FkP’xy (Cx)
7. Y: Compute Cx = FkP’xy (Cx)/FkP’xy; Generate unique challenge Cy
8. Y→X: FkP’xy (Cx, Cy).
9. X: Decrypt message sent by Y to obtain Cx and Cy. Compare former with his own challenge, if they match, go to next step, else abort.
10. X→Y: FkP’xy (Cy).
11. Decrypt message from X and compare with challenge Cy. If they match, authenticate, and encrypt subsequent messages with FkP’xy.

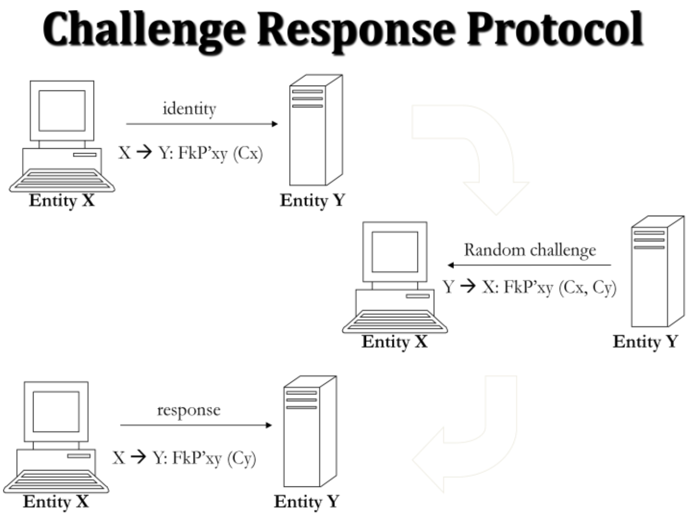

Figure 2.0 Encryption with Challenge Response Protocol

In step 1, X generates a public / private key pair (Ex(Fk), Dx(Fk)) and also derives a secret key P’ from the password *p*. In step 2, X encrypts the public key Ex(Fk) with P’ and sends it to Y. In steps 3 and 4, Y decrypts the message using the stored password of X and uses Ex(Fk) with P’ to encrypt a XORed key FkP’xy and sends it to X. In steps 5 and 6 X uses this XORed key to encrypt a unique challenge Cx and sends it to Y. Y decrypts the message to obtain the challenge and generates a unique challenge of its own Cy. In step 8, Y encrypts Cx and Cy with FkP’xy and sends it to X. X verifies Cx and decrypts Cy, which it sends back to Y. In step 11, Y decrypts this message and compares it to its own challenge Cy. If they match, the protocol is successfully completed and subsequent messages from both entities are encrypted by FkP’xy.

3.5 TERMINATION

After the authentication of both the users (client/ server), the secured and authenticate way is provided for the message exchange. Every time a user wants to send a message to another user, process (3.4) is executed and when the entities at both the ends are authenticated the start of message transaction takes place after finishing off the password and the keys are cycled off to new values.

4. Conclusion

The paper describes the integrated mechanism of authentication using challenge response protocol with password application. There has been the use of asymmetric key transfer for password authentication. The proposed scheme reduces the risk of third party attacks it also proves the unique identity of user pairs by one to one association has the data transaction over the millions of users participating in network communication becomes properly secure.

5. FUTURE SCOPE

The present work has been an eminent approach towards the security aspects of message transaction. The scheme can be further implemented by newly registered client/server pairs. The password based authentication on user keys for secured exchanged in future technologies. The large amount of Chat-servers, social sites require intrusion free data transfer so that better & efficient user interactions can be implemented.

The new algorithms are based on the schematic viewport of the data users. The continuous increase in internet users has led to a more secured data communication scheme which the present paper implies. There is also a margin of betterment in present scheme that is to make the password exchange more encapsulated, isolated and confidential. It can be further improved as per the core demands. The client user session can also be included to avoid noise interference of external agents thus it can be observed that the method is readily acceptable for future work.

REFERENCES

[1] Arkko J, et al. Security mechanism agreement for SIP sessions. IETF Internet draft, June 2002.

[2] Franks J., Hallam-Baker P., Hostetler J., Lawrence S. HTTP authentication: Basic and digest access authentication, 2617, IETF Network Working Group, June 1999.

[3] Rosenberg J., Schulzrinne H., Camarillo G., Johnston A., Peterson J., Sparks R., Handley M. and Schooler E. SIP: Session Initiation Protocol. RFC 3261, IETF. The Network Working Group, June 2002.

[4] Lin, C. L. and T. Hwang. “A password authentication scheme with secure password updating,” Computers and Security, vol. 22, no. 1, pp. 68-72, 2003.

[5] M. Bellare, A. Boldyreva, A. Desai, D. Pointcheval, Key-privacy in public-key encryption, in: Asiancrypt ’01, Lecture Notes in Computer Science, Vol. 2248, Springer, Berlin, 2001, pp. 566–582.

[6] M. Bellare, D. Pointcheval, P. Rogaway, Authenticated key exchange secure against dictionary attacks, Advances in Cryptology—Eurocrypt’ 2000, Lecture Notes in Computer Science, Vol. 1807, Springer, Berlin, 2000, pp. 139–155.

[7] M. Bellare, P. Rogaway, The AuthA protocol for password-based authenticated key exchange, Contributions to IEEE P1363.2 working group.

[8] CRAMER, R.ANDSHOUP, V. 1998. A practical public key cryptosystem provably secure against adaptive chosen cipher text attack. In Advances in Cryptology—CRYPTO ’98, H. Krawczyk,Ed. Springer-Verlag, New York, 13–25.

[9] DIERKS, T.ANDALLEN, C. 1999. The TLS protocol: Version 1.0. Request for Comments:2246. ftp://ftp.isi.edu/in-notes/rfc2246.txt.DIFFIE, W., VANOORSCHOT, P. C., ANDWIENER, M. J.

1992.Authentication and authenticated key exchanges.Des. Codes Cryptography 2, 2 (June 1992), 107–125.

[10] S. Bellovin, M. Merritt, Encrypted key exchange: password-based protocols secure against dictionary attacks, Proc. 1992 IEEE Computer Society Conf. on Research in Security and Privacy, 1992, pp. 72–84

[11] C. Boyd, P. Montague, K. Nguyen, Elliptic curve based password authenticated key exchange protocols, in: ACISP '01, Lecture Notes in Computer Science, Vol. 2119, Springer, Berlin, 2001, pp. 487–501.

[12] E. Bresson, O. Chevassut, D. Pointcheval, Security proofs for an efficient password-based key exchange, Proc. 10th ACM Conf. on Computer and Communications Security, 2003, pp. 241–250.