International Journal of Advance Research in Science and Engineering
Vol. No.5, Special Issue No. (01), February 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

RESEARCH STUDY ON TWO FACTOR ZERO KNOWLEDGE PROOF AUTHENTICATION SYSTEM

Niranjanamurthy M [1], Shashank K S[2], Sumanth P Gowda[3], Suhas Bhatta S[4]

[1,2,3,4]*Department of Computer Applications, MSRIT, Bangalore, (India)*

ABSTRACT

Two Factor Zero Knowledge Proof Authentication System" is a system which is a combination of two techniques namely "Two Factor Authentication" technique and "Zero Knowledge Proof" technique to provide security for the user to login to the secured system from the untrusted device. Two Factor Authentication is a method commonly used by internet services to provide an extra layer of security in addition to the standard password used as login credentials. It employs a secondary device, such as a phone that the user must have in his or her possession to complete the authentication process.In services such as Google's Gmail, two factor authentication works using a phone's SMS capabilities to send text messages from authentication servers with access codes to the phone. This is combined with the password, so authentication requires both knowledge and possession of a trusted device.Two factor authentication still requires sending a password over a network, however, to combat this, the concept of zero knowledge proof have been used in the system. This paper we discussed Two Factor Authentication, Properties of Zero Knowledge Proof, Working of the System, Stages involved in the Zero Knowledge, Architecture, Advantages

Keywords: Two Factor Authentication, Properties of Zero Knowledge Proof, Working of the System, Stages involved in the Zero Knowledge, Architecture, Advantages

I. INTRODUCTION

Two Factor Authentication is a method commonly used by internet services to provide an extra layer of security in addition to the standard password used as login credentials. It employs a secondary device, such as a phone that the user must have in his or her possession to complete the authentication process.

Zero knowledge proof or protocol is method in which a party A can prove that given statement X is certainly true to party B without revealing any additional information. Zero-knowledge authentication protocols are an alternative to authentication protocols based on public key cryptography. Low processing and memory consumption make them especially suitable for implementation in smart card microprocessors, which are severely limited in processing power and memory space. This paper describes a design and implementation of a software library providing smart card application developers with a reliable authentication mechanism based on well-known zero-knowledge authentication schemes. Java Card is used as the target smart card platform implementation based on the evaluation of the Fiat-Shamir (F-S) and Guillou-Quisquater (G-Q) protocols under various performance criteria are presented to show the effectiveness of the implementation and

that G-Q is a more efficient protocol.The Zero Knowledge Proof (ZKP) authentication protocol is used in cryptography to allow a party to prove that he/she knows something (e.g. a credential), without having to transmit this credential. There are two parties involved in ZKP; the prover A and the verifier B, where ZKP enables a "prover" to show that they have the credential (ie, credit card number or password), without having to give the "verifier" the credential details. With ZKP there is no transmission or storage of password /credential details on the authentication server. ZKP delivers the following benefits:

Zero-Knowledge: if the statement is true, the verifier will not know anything other than that the statement is true. Information about the details of the statement will not be revealed.

Completeness: If the statement is true, the honest verifier (that is one following the protocol properly) will be able to prove that the statement is true every time.

Soundness: if the statement is false, it is almost impossible to an astronomically small chance that someone could fake the result to the verifier that the statement is true.

Two factor authentication still requires sending a password over a network, however, to combat this, the concept of zero knowledge proof have been used in the system. The Zero-Knowledge Proof is a method used in many cryptography systems. It allows a party to prove that he/she knows something (i.e. credential), without having to send over the value of the credential. In this implementation, it will be used to prove the password of the user without sending over the actual password. The system also allows for no password hashes to be stored on the server.

Properties of Zero Knowledge Proof

a) Completeness: If the statement is true, the honest verifier (that is, one following theprotocol properly) will be able to prove that the statement is true to an honest verifier every time.

b) Soundness: If the statement is false, it is not possible (with a very small chance) to fake the result to the verifier that the statement is true.

c) Zero-knowledge: If the statement is true, the verifier will not know anything other than that the statement is true. Information about the details of the statement will not be revealed.

Principles used in the Scheme:

a) User: The user is the individual who wishes to securely maintain and access an account with one or more services.

b) Trusted Device (TD): The trusted device is a device that belongs to the user. It must be a mobile device such as a phone, tablet, or laptop because it will be required every time the user attempts to log in. We assume that the trusted device is not compromised.

c) Untrusted Device (UD): An untrusted device is a device that the user wishes to use in order to log into an account that the user has. The untrusted device is, as the name implies, not trusted; it is assumed to be key-logged and shoulder surfed.

d) Server: A server in this scheme is the device in charge of a particular service. All interaction with the service is achieved through communication with the server.

e) Adversary: In this scheme we consolidate all possible malicious actions under a single hypothetical individual who we refer to as the adversary.

II. RELATED WORK

Authentication systems have highly evolved in recent years, particularly in public environments especially in web applications. Also, activities and government enter-prizes rely increasingly on these technologies. This protocol requires deification by username/password and monitoring states of sessions and cookies. In addition, their facilities implementation and deployment have made omnipresent and unavoidable. Their seductive and op-opportunities in the evolution of companies encourage more attackers to re-evolve their ways of attacks.[1] More critical, the consciousness and the behaviour of the users have very remarkable influences on the survival of their accounts. But, it is impossible to see them as a key of safety at the level university. Our contribution comes in the optics to insure and to create symmetric secure communication channel between the clients and the Web server. The interest, is to have a dynamic identification system which combines three approaches. The first one insures the regeneration of the virtual passwords, and the confidentiality and the integrity of the nonces of mutual authentication exchanged. The second calculates the secret session key shared between the client and the web server.[2]

Technology	Pros	Cons
Javascript Each website could have a JavaScript script which automatically processes the challenge, username, and password to produce a response.	• Seamless integration • Transparent to user • Supported by all modern browsers • Excellent mobile support	• Sometimes disabled for security reasons • Requires more trust in the website
Applets This is the approach used by the original paper, which used an applet to download a python script which performed the authentication. We believe this solution is strictly inferior to using JavaScript.	• No extra work from user • Supported by all modern browsers	• Requires a plugin to be installed • Very little mobile support • Requires more trust in website
Browser Extension A browser extension could provide an interface that allows the user to easily calculate the response from a challenge, either by manual form entry or by scraping the web page.	• Can easily be verified as secure (it can only be changed with the client's knowledge) • Can integrate nicely with webpage	• Potentially requires more work by user • Needs to be installed for every browser and computer
Client Side Script This is an alternative to a browser extension; the user could download a script or program that they can run to manually calculate the response from the challenge. This would be on either a computer or a phone that the user will have easy access to.	• Can easily be verified as secure	• Requires more work by user • Needs to be installed on a device that can be easily accessed whenever the user wants to log in.
Third Party Website A trusted third party website could function as a service that calculates responses from challenges.	• Portable • Easily accessible	• Requires more work by user • Central point of failure.

[3]

In many applications, the password is sent as cleartext to the server to be authenticated thus providing the eavesdropper with opportunity to steal valuable data. This paper presents a simple protocol based on zero knowledge proof by which the user can prove to the authentication server that he has the password without having to send the password to the server as either cleartext or in encrypted format. Thus the user can authenticate himself without having to actually reveal the password to the server. Also, another version of this

protocol has been proposed which makes use of public key cryptography thus adding one more level of security to the protocol and enabling mutual authentication between the client & server.[4]

Efficient zero-knowledge proofs of knowledge (ZK-PoK) are basic building blocks of many practical cryptographic applications such as identication schemes, group signatures, and secure multiparty computation. Currently, rst applications that critically rely on ZK-PoKs are being deployed in the real world. The most prominent example is Direct Anonymous Attestation (DAA), which was adopted by the Trusted Computing Group (TCG) and implemented as one of the functionalities of the cryptographic Trusted Platform Module (TPM) chip. Implementing systems using ZK-PoK turns out to be challenging, since ZK-PoK are, loosely speaking, signicantly more complex than standard crypto primitives, such as encryption and signature schemes. As a result, implementation cycles of ZK-PoK are time-consuming and error-prone, in particular for developers with minor or no cryptographic skills.[5]

A security concern regarding the authentication protocols is that of the malicious verifiers. A malicious veri er poses herself as an honest verier, engages in the protocol, deviates from the protocol,and tries to gain knowledge about the secret stored on the smart card. While bilateral authentication protocols may help by aborting the protocol in case one of the parties fails to authenticate herself to another, it does not prevent partial leakage of information. The leakage might be undesirable for systems which require a high level of security.[6]. Zero-knowledge proofs are cryptographic protocols which do not disclose the information or secret itself during the protocol. Zero-knowledge proofs plays an important role in the design of cryptographic protocols. The application of Zero-knowledge protocols can be in authentication, identification, key exchange and other basic cryptographic operations. Zero-knowledge proof has been implemented without expose any secret information during the conversation and with smaller computational requirement than using comparable public key protocols.[7]. Zero Knowledge Protocols, is an improvement on these situations. The objective is to obtain a system in which it is possible for a prover to convince a verifier of his knowledge of a certain secret without disclosing any information. The present invention relates to Zero Knowledge Protocols that allows the knowledge of some “secret” or private key information in a first party domain to be verified by a second party without imparting the actual secret information or private key to that second party or to any eavesdropping third party. Throughout the present specification, the first party owning the secret information or private key (“s”) and wishing to prove that it has possession of the information will be referred to as the “prover” (“P”); the second party wishing to verify that this is the case without actually receiving knowledge of the secret will be referred to as the “verifier” (“V”). The prover P and verifier V may be any suitable electronic device. The secret information may be any numeric value, hereafter referred to as the secret number of the prover P. ZKP based protocols require less bandwidth, less computational power, and less memory compared to other authentication methods and thus seems to be suitable for WSN.[8]

Mechanism Of ZKP : In WSN using ZKP one party assures another that a statement is true instead of showing anything other than the veracity of the statement. The prover and the verifier uses some numeric value, which acts as secret number for prover. Basically computational intensive mathematical problem are normally offered by prover p, and many possible solutions to this problem are normally requested from verifier side. If critical information relating to the solution is knows by p then replay with any one requested available solutions to it. If

the P not knows anything related with critical information, then he is unable to provide the needed information to the V.[9]

Key mathematical knowledge applied to zero-knowledge proof protocol: The commonly used algorithm theory in zero-knowledge proof , which is similar to public-key cryptosystems in cryptography ,is mainly based on the following key mathmetic problem:

(1)the square root problem of modulo n Given a positive integer n , a∈Zn, if there exists x∈Zn that makes x2=a(mod n), then x is called as a square root of modulo n.

(2) the calculation problem of the discrete logarithm Given a prime number p,and a,which is one of the primitive element on finite field Zp .To b on Zp,looking for one and only integer c that makes a^c≡b(mod p). In general, the problem is difficult if you are looking forward p, and there is still no algorithm to calculate polynomial of discrete logarithm. The method based on the elliptic curve discrete logarithm is commonly used.

(3) Large integer factorization problem The factorization of a large integer M which is N digit, is usually impossible to be done in O (N) , but rather to up to O (exp (N)) level. [10]

Zero Knowledge: It means that if the statement is true, no cheating verifier learns anything other than this fact.The aim of the zero knowledge is to prove the knowledge of a secret without revealing it. Each user from the group has a secret information and each one has to prove that he/she knows the information without revealing it to the server. Thus the prover is the user and the verifier is the server. Since the secret information of each user is different therefore the server will identify each user through a demonstration of his knowledge. The basic idea of the zero knowledge authentication is that the verifier asks a question related to the secret information in such a way that the answer does not reveal the secret. Schnorr's protocol [19] is one of the most popular zero knowledge protocol. Let p and q be two primes number such that q divides $(p 1)$. Let g 1be an element of order q in $_p Z$ (the multiplicative group of integers modulo p). Also let $_q G$ be the cyclic subgroup of order q generated by g . The integers p,q, g are known and can be common to a group of users. An identity consists of a private /public key pair. The private key w is a random non-negative integer less than q . The public key is computed as $y g p _w$ mod. [11]

Zero-knowledge authentication protocols are an alternative to authentication protocols based on public key cryptography. Low processing and memory consumption make them especially suitable for implementation in smart card microprocessors, which are severely limited in processing power and memory space. This paper describes a design and implementation of a software library providing smart card application developers with a reliable authentication mechanism based on well-known zero-knowledge authentication schemes. Java Card is used as the target smart card platform implementation based on the evaluation of the Fiat-Shamir (F-S) and Guillou-Quisquater (G-Q) protocols under various performance criteria are presented to show the effectiveness of the implementation and that G-Q is a more efficient protocol.[12]

III. STAGES INVOLVED AND WORKING OF THE SYSTEM

There are two stages involved in the system. They are,

a) Signup Stage/ Account Creation: This is the stage in which the user creates an account with a website or other service.

b) Login Stage: This stage consists of the steps necessary for the user to log in.

Working of the System:

Two Factor Authentication:

a)Account Creation: Before the user can log into an account with a service, the user must have an account with that service. To begin, the user selects a username and communicates it to the server via the trusted device. If this username is already taken, the user is informed with an error message, and the user must make another attempt. When the user identifies an unused username,the process can continue.

In the next step, the user selects a password and enters it into the trusted device. The trusted device uses that password to generate a secret key. The trusted device then randomly generates and saves a secret key for itself. The public keys associated with each of the two secret keys are then computed. After that, the username and both public keys are sent to the server, which stores this data. Once this process is complete, the trusted device deletes all references to the user's password or to the associated public key This set of steps is summarized in figure 1 below..

Figure 1 - Summarized account creation process

b) Login Stage: The user, having created an account may find that he wishes to log in from an untrusted device. He begins by selecting the service that he wishes to log into on his trusted device. The device sends a message, digitally signed with the device's secret key, to the server indicating the intent of the user (as identified by his username) to log in. The server saves in its records that a login attempt has begun. Just in case, this login attempt expires within a minute if it is not continued. The steps so far have been summarized in figure 2.

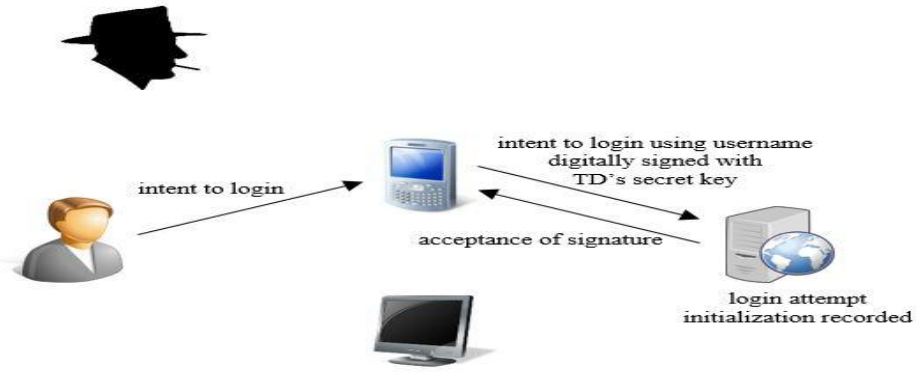

Figure 2 - Summarized initial login process

After that, the trusted device requests that the user enter his password. Once the password is entered, the trusted device computes the user's secret key from the password. Next, the trusted device runs through several rounds of

zero knowledge proof with the server to demonstrate knowledge of the user's secret key to the server. Every message sent should be signed with the trusted device's secret key such that all messages not sent by this device can be ignored.When the server is sufficiently convinced that the trusted device currently knows the password, it generates a random token and associates that token with this login attempt. Once again, if the token remains unused for too long, the login attempt expires. The token is encrypted with the trusted device's public key and sent to the trusted device. The trusted device decrypts the message and displays the token on screen. The user then enters his username and the token into the untrusted device which sends the data on to the server. These steps are shown in figure 3

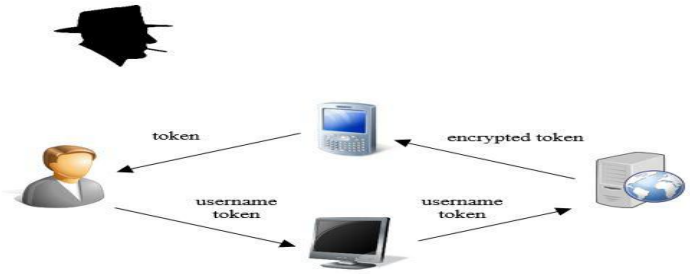

Figure 3 –Summarized halfway login process

The server checks that the user has a login attempt in progress and verifies that the token entered matches the one associated with that login attempt. If everything checks out, the untrusted device is informed that it successfully logged in half way, and the token is removed from the records as a result, the token can only be used once.

The untrusted device then displays to the user whether or not he has logged in half way. The user uses this information to inform the trusted device of whether the first half of logging in was successful. The device in turn, forwards this answer to the server. If the answer is no, the login attempt is aborted and must be restarted. If the answer is yes, the untrusted device must be the single device that is logged in half way, and the protocol can continue. These steps are summarized in figure 4.

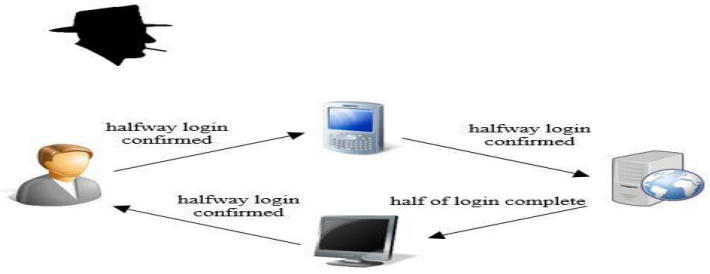

Figure 4: Summarized halfway login confirm process

At this point, the server sends another encrypted token. Once again, the trusted device decrypts the token and the user copies it to the untrusted device. This time, when the untrusted device sends the data to the server, the server verifies not only that the token is correct, but also that the device attempting to log in is the unique device that is half way logged in. At that point, the device's halfway login is upgraded to a full login, and the server can serve the untrusted device the content that the user is attempting to access. Once the second to- ken is printed on

screen, the trusted device removes all record of the user's password or secret key.These steps are summarized in figure 5.

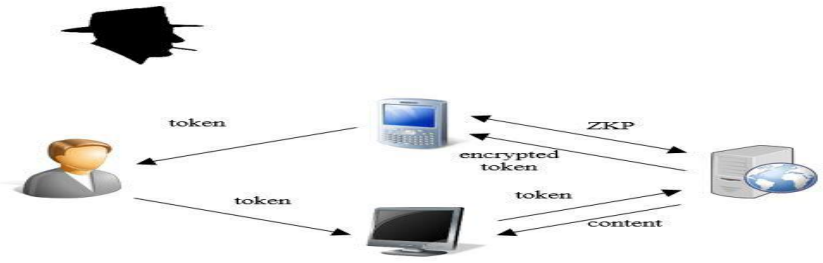

Figure 5: Summarized final login process

IV.ZERO KNOWLEDGE PROOF

The components used in the Zero Knowledge Proof algorithm are:

G:This is a cyclic group. This group contains a set of numbers which is based ona formula. This is a public group which will be available to both prover (user) and verifier (server).

g0:A generator of the group G. It is an element of the group G. This is a public variable which will be available to both prover (user) and verifier (server).

X:The hash of the password that the user inputs.

Y:The pseudonym of the user. This is used for the verifier in the calculation of the proof of knowledge.

a: The random token generated for each login attempt.

T1,rx,zx, c: Other miscellaneous variables which are used in the calculation.

1) Stages involved in the Zero Knowledge Proof System:

There are three stages involved in the Zero Knowledge Proof System, they are:

a) Initialization.

b) Registration Process.

c) Authentication Process.

a) Initialization.

This is where the public key is created. Simply, the value g0 is generated.

1.Given group G. Let g0, g1 be random elements of G.

2. Let the public key be zkapk ={G, g0}.

b) Registration Process.

This is where the pseudonym of the user is created. This is simply done by hashing the user's password into x, and calculating Y=g0x.

1. User inputs username and password.

2. The user hashes the password with Hash function, H and calculates x H(password).

3. The user then computes Y = g0x

4. The user sends (username, Y) to the server

5. The server stores (username, Y) into the database.

c)Authentication Process

1. The server generates a random one-time token a and stores it and sends it to the user.
2. User inputs username and password.
3. The user hashes the password with Hash function, H and calculates
x = H(password).
4. The user then computes Y = g0x.
5. The user generates random rx _ G and calculates T1=g0rx.
6. The user then calculates c=H(Y,T1,a).and Zx = Rx – cx.
7. The user sends (c, Zx) over to the server.
8. The server calculates T1=Ycg0zxand verifies that c=H(Y,T1,a).
9. If successful, user is authenticated.
10. The server generates a random one-time token a and stores it and sends it to the user.
11. User inputs username and password.
12.The user hashes the password with Hash function, H and calculates x = H(password).
13. The user then computes Y = g0x.
14. The user generates random rx _ G and calculates T1=g0rx.
15. The user then calculates c=H(Y,T1,a).and Zx = Rx – cx.
16. The user sends (c, Zx) over to the server.
17. The server calculates T1=Ycg0zxand verifies that c=H(Y,T1,a).
18. If successful, user is authenticated.

No	User (Prover)		Verifier (Server)
1			Generate random **a**
2	Receive a	←	Send a
3			
4	Calc. x=H(password)		
5	Calculate $Y=g0^x$		
6			
7	Randomly generate rx		
8	Calculate $T1=g_0rx$		
9	Calculate c=H(Y,T1,a)		
10			
11	Calculate $z_x=r_x - cx$		
12	Send c, z_x	→	Receive c, z_x
13			Calculate $T1=Y^c g_0^{zx}$
14			Check if c= H(Y,T1,a)

Table 1– Authentication Process

If we look at the formula of T1 in step 8 and step 13:

Step 8: T1=g0rx

Step 14: T1=Ycg0zx

We will have to prove that: g0rx= Ycg0zx

With reference to step 5 and step 11, we know that:

Step 5: Y=g0x

Step 11: zx=rx – cx

Therefore, by performing a simple substitution, we can prove that:

g0rx= Yc g0zx

g0rx= (g0x)c g0(r x-cx)

g0rx= g0cx g0r x-cx

g0rx= g0cx + r x-cx

g0rx= g0r x(Proven)

With all 3 elements, we can verify that c=H(Y,T1,a), thus proving that user knows x

V. ZERO KNOWLEDGE PROOF ARCHITECTURE

a) Registration Process

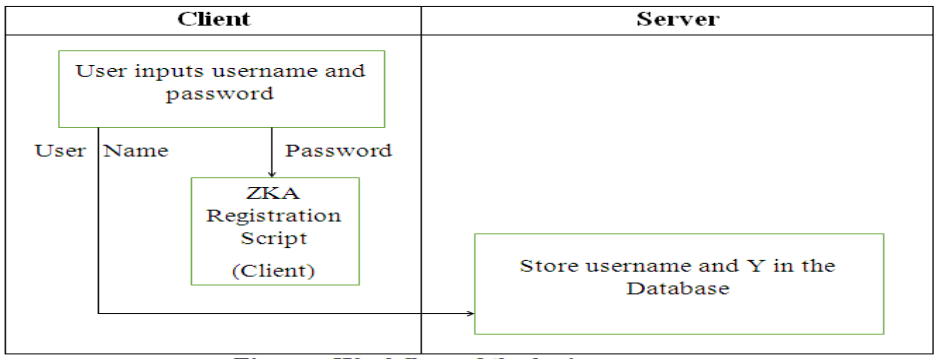

Figure - Workflow of the login process

Based on this registration process, we can see that the password hash or password is not stored on the database at all. Even though attackers manage to obtain the username and Y, they are not able to compute the password hash, x.

b) Authentication Process

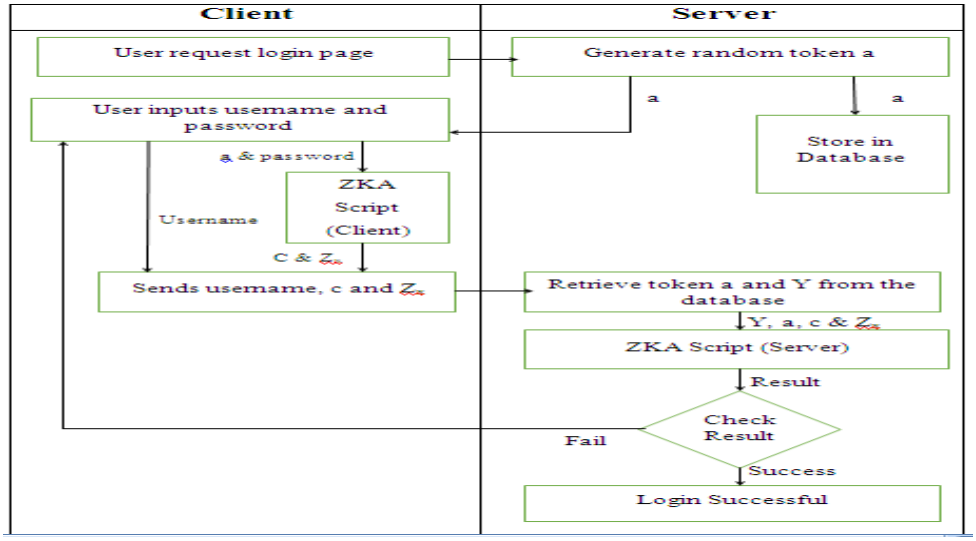

Figure 6 - Workflow of the authentication process

The user firsts requests for the login page. This page will then pass back a one-time token a, which is stored in the server database (for that session), and passed back tothe user. This will be used in the hash later on to prevent using the same, valid credentials.From this we can also see that whatever is sent over, c &Zx, has no meaning to the attacker. Even if he understands the formula, it is impossible to reverse a hash and obtain x from

Zx. Thus, we can see that the user can easily prove his/her identity without having to send over any confidential data.

Advantages:

a) No additional hardware required to implement this system.

b) Data transmitted through the network is useless to the attackers.

c) Through the use of a one-time token in the hashing function, the information sent over is only valid for once, and thus will not be usable by attackers who intercept the information.

d) Two factor authentication ensures more security.

V. CONCLUSION

Two Factor Authentication is a method commonly used by internet services to provide an extra layer of security in addition to the standard password used as login credentials. It employs a secondary device, such as a phone that the user must have in his or her possession to complete the authentication process.

Besides providing a much higher level of security to a web application, there are many other reasons why "Two Factor Zero Knowledge Proof Authentication" is worth implementing. Firstly, it allows for someone with no knowledge on how the protocol works taking advantage of such a concept. Also the system is basically transparent to the user. The two factor authentication provides an extra layer of security. Finally, the simplicity and ease to implement the system is definitely a value-add and a reason to have this in any web application. No additional hardware required to implement this system.Data transmitted through the network is useless to the attackers. Through the use of a one-time token in the hashing function, the information sent over is only valid for once, and thus will not be usable by attackers who intercept the information. Two factor authentication ensures more security.

VI. ACKNOWLEDGEMENTS

I thank Dr. T. V. Suresh Kumar, Prof. and Head, Dept. of MCA, MSRIT, Bangalore-54. for his continuous support and encouragement for completing this research paper and also thanks to MSRIT management.

I thank Mr. Chethan Venkatesh, Assistant Professor. of Dept. of MCA, MSRIT, Bangalore-54, for his valuable guidance and support for completing this paper.

REFERENCES

[1] Younes Asimi, Abdallah Amghar, Ahmed Asimi and Yassine Sadqi-"Strong Zero-knowledge Authentication Based on Virtual Passwords"International Journal of Network Security, Vol.18, No.4, PP.601-616, July 2015

[2] Younes ASIMI Abdellah AMGHAR Ahmed ASIMI and Yassine SADQI-"STRONG ZERO-KNOWLEDGE AUTHENTICATION BASED ON THE SESSION KEYS (SASK)" International Journal of Network Security & Its Applications (IJNSA) Vol.7, No.1, January 2015

[3] RyanCheu,Patrick Yang, Alexander Lin, Alexander Jae- "An Implementation of Zero Knowledge Authentication" NARWHAL Massachvsetts Institvte of Technology May 14, 2014

[4] DATTA-"ZERO KNOWLEDGE PASSWORD AUTHENTICATION PROTOCOL" International Journal of Communication Network Security ISSN: 2231 – 1882, Volume-1, Issue-4, 2012

[5] Endre Bangerter, Stefania Barzan, Stephan Krenn2,Ahmad-Reza Sadeghi3, Thomas Schneider3, and Joe-Kai Tsay3 -"Bringing Zero-Knowledge Proofs of Knowledge to Practice" FP7 EU project CACE (Computer Aided Cryptography Engineering)

[6] Mohammad Sadeq Dousti and Rasool Jalili "Effcient Statistical Zero-Knowledge Authentication Protocols for Smart Cards Secure Against Active & Concurrent Attacks" An abridged version of this paper is published in International Journal of Computer Mathematics, January 2015

[7] Jitendra Kurmi, Ankur Sodhi "A Survey of Zero-Knowledge Proof for Authentication" International Journal of Advanced Research in Computer Science and Software Engineering Volume 5, Issue 1, January 2015

[8] Vishal Parbat, Tushar Manikrao,Nitesh Tayade, Sushila Aghav "Zero Knowledge Protocol to design Security Model for threats in WSN" (IJERA) ISSN: 2248-9622 Vol. 2, Issue 2, pp.1533-1537, Mar-Apr 2012

[9] Manish P.Gangawane "Implementation Of Zero Knowledge Protocol In Wireless Sensor Network for Identification Of Various Attacks" International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 8, August 2012

[10] Wang Huqing, Sun Zhixin - "Research on Zero-Knowledge Proof Protocol"IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 1, January 2013.

[11] Kumar - "A Secure And Efficient Authentication Protocol Based On Elliptic Curve Diffie-Hellman Algorithm And Zero Knowledge Property" (IJSCE)ISSN: 2231-2307, Volume-3, Issue-5, November 2013

[11] Ahmed Patel, Kenan Kalajdzic, Laleh Golafshan, Mona Taghavi - "Design and Implementation of a Zero-Knowledge Authentication Framework for Java Card" International Journal of Information Security and Privacy, 5(3), 1-18, July-September 2011