The Weaknesses of the Virtual Password Authentication Protocol with Cookie

Hsieh-Tsen Pan[1], Chia-Chun Wu[2], Cheng-Ying Yang[3] and Min-Shiang Hwang[1,4,*]

[1]Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan 41354
[2]Department of Industrial Engineering and Management, National Quemoy University, Taiwan
[3]Department of Computer Science, University of Taipei, Taipei, Taiwan (Email: cyang@utaipei.edu.tw)
[4]Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan 40402
(*Email: mshwang@asia.edu.tw)

Abstract. Password-based authentication protocols are susceptible to various attacks. Recently, Sood, Sarje, and Singh proposed an inverse cookie-based virtual password authentication protocol. Their protocol is practical and easy to implement. They claim that their scheme is secure to against various attacks, include online dictionary attack, offline dictionary attack, eavesdropping attack, denial of service attack, phishing attack, pharming attack, man-in-the-middle attack, replay attack, leak of verifier attack, message modification or insertion attack, and brute force attack. However, we find that some weaknesses of Sood *et al.*'s scheme. In this article, we will show that Sood *et al.*'s scheme is vulnerable to the on-line guessing password attack and the denial of service attack.

1. Introduction

The password authentication systems are wide spread in authentication technique to authenticate the remote legal users [1-4]. However, the Password-based authentication protocols are susceptible to various attacks such as dictionary, guessing, phishing, stolen verifier, denial of service, man-in-the-middle, insider attacks, and other attacks [5-11]. It's an important work to prevent a password from being compromised [12-20].

In 2001, Hwang et al. proposed an improvement of the SPLICE/AS authentication system in WIDE (Widely Integrated Distributed Environment) [21]. Their method could against to the guessing password attack. In 2016, Amin proposed an ID-based user authentication scheme for multi-server environment. They claimed that his scheme resisted various possible attacks include off-line identity guessing attack, off-line password guessing attack, and smart card stolen attack [22]. However, Pan, Tsaur, and Hwang found his scheme was vulnerable to off-line identity guessing with smart card stolen attack and off-line password guessing with smart card stolen attack [23]. In order to enhance the security of user authentication scheme, many smart card-based user authentication schemes had been proposed [24-33].

Recently, Sood, Sarje, and Singh proposed an inverse cookie-based virtual password authentication protocol [34]. The virtual password helps to against the different types of attacks for password authentication protocols. Their protocol is practical and easy to implement. They claim that their scheme is secure to against various attacks, include online dictionary attack, offline dictionary attack, eavesdropping attack, denial of service attack, phishing attack, pharming attack, man-in-the-middle attack, replay attack, leak of verifier attack, message modification or insertion attack, and brute force attack. However, we find that some weaknesses of Sood et al.'s scheme. In this article, we will show that Sood et al.'s scheme is vulnerable to the on-line guessing password attack and the denial of service attack.

For more details we divide this paper into 4 Sections as follows: In Section 2, we briefly review Sood et al.'s virtual password authentication protocol with cookie. In Section 3, the vulnerabilities of their protocol are analysed. Finally, we make a conclusion of the paper in Section 4.

2. Review of Sood *et al.*'s Scheme

In this section, we briefly review the second protocol, virtual password authentication protocol with cookie [34]. In the scheme, there are two participants, the user U_i and the server S. There are three phases in their scheme: The registration, login, and authentication phases. Some notations used in the scheme are described in Table 1.

Table 1. The notations of Sood *et al.*'s scheme.

Notation	Meaning
U_i	The *i*th user
S	The server
ID_i	The identity of U_i
P_i	The password of U_i
URL	A URL of the destination server.
OTP	A one-time password of server for each client
MAX_TRUST	A maximal trust assigned to U_i
MIN_TRUST	A minimal trust assigned to U_i
Current_TRUST	A current trust value of U_i
PK	The server's public key
SK	The server's private keyy
SS	A session key of SSL protocol
$H(\cdot)$	A one-way hash function
$\oplus$	An exclusive-or operation
$\|\|$	A concatenation operation

2.1. Registration Phase

A new user U_i has to register to a server S to be a legal user. The user U*i* registers to the server S by the following steps:

Step 1. The U_i submits his/her identity IDi and password Pi to the server S over a secure channel.

Step 2. After receiving the registration **request** message from the U_i, the server S computes and stores {IDi, Ai, MIN_TRUST, MAX_TRUST, CUR_TRUST, CK, Ti} in its database:

$$A_i = Pi \oplus SK \oplus OTP,$$
$$CK = H(Ns \,||\, URL \,||\, PK),$$
$$Ti = OTP \oplus H(SK).$$

Here, OTP and Ns are two random numbers which generated by the server; PK and SK are the server's public key and private key, respectively; URL is the server's Uniform Resource Locator.

Step 3. The server S sends the cookie information CK to the user Ui over a secure channel.

Step 4. The user U_i stores the cookie information CK into their terminal device.

2.2. Login Phase

The user Ui wants to have the service of the server S. The user Ui needs to executes and sends a login request message to S as follows:

Step 1. The U_i chooses a session key SS and sends $E_{PK}(SS)$ to the server. Here, $E_{PK}(SS)$ is the encrypts the session key SS with the server's public key.

Step 2. The user Ui submits his/her identity IDi and password Pi to his/her terminal with the browser. If the user Ui's browser contains a cookie CK, the user Ui's browser computes Ki and sends {Ki, CK} to the server:

$$Ki = H(Idi \,||\, URL \,||\, PK \,||\, Pi \,||\, SS \,||\, CK).$$

2.3. Authentication Phase

Upon receiving the login request message {Ki, CK} from the user Ui, the server S authenticates the user as follows:

Step 1. The server S retrieves the user Ui information in the server's database by CK. If it's not CK data in the database, the server rejects the login request.

Step 2. The server S compared the CUR_TRUST and MIN_TRUST. If CUR_TRUST larger or equal to the MIN_TRUST, the server computes OTP, Pi, and K`I as follows:

$$OTP = T_i \oplus H(SK),$$
$$P_i = A_i \oplus SK \oplus OTP,$$
$$K'_i = H(ID_i \,||\, URL \,||\, PK \,||\, P_i \,||\, SS \,||\, CK).$$

Step 3. The server S verifies K`I whether is equal to Ki. If it's not equal, the server rejects the login request. Otherwise, the server proceeds to the next step.

Step 4. The server S computes Mi and Qi as follow:

$$M_i = Nk \oplus H(Idi \,||\, SS \,||Pi),$$
$$Q_i = H(ID_i \,||\, Nk \,||\, P_i \,||\, SS),$$

where Nk is a random nonce which generated by the server.

Step 5. The server S sends the mutual authentication message $\{M_i, Q_i\}$ to the user U_i.

Step 6. Upon receiving the mutual authentication message $\{M_i, Q_i\}$ from S, the user Ui computes N`k and Q`I as follows:

$$N`k = Mi \oplus H(Idi \,||\, SS \,||Pi),$$
$$Q`i = H(IDi \,||\, N`k \,||\, Pi \,||\, SS),$$

The user checks Q`I whether is equal to Qi. If it holds, the user and server compute the common session key SK = H(SS || Pi || Nk || CK || Idi). Otherwise, the session is terminated.

3. Cryptanalysis of Sood et al.'s Scheme

In this section, we will show that some weaknesses in Sood et al.'s virtual password authentication protocol with cookie: On-line guessing password attack and denial of service attack.

- On-line Guessing Password Attack:

  We assume that an adversary is able to operate a legal user's computer in some reasons.

  Step L1. The adversary chooses a session key SS and sends $E_{pk}(SS)$ to the server. Here, $E_{pk}(SS)$ is the encrypts the session key SS with the server's public key.

  Step L2. The adversary submits Ui's identity IDi and guesses Ui's password P`i to Ui's terminal with the browser. If the user Ui's browser contains a cookie CK, the user Ui's browser computes Ki and sends {Ki, CK} to the server:

  $$Ki = H(IDi \,||\, URL \,||\, PK \,||\, P`i \,||\, SS \,||\, CK).$$

  Upon receiving the login request message {Ki, CK} from the adversary, the server S authenticates the user as follows:

  Step A1. The server S retrieves the user Ui information in the server's database by CK. If it's not CK data in the database, the server rejects the login request. Since the CK is stored in the user's client browser, therefore the server will authenticate the adversary.

  Step A2. The server S compared the CUR_TRUST and MIN_TRUST. If CUR_TRUST larger or equal to the MIN_TRUST, the server computes OTP, Pi, and K`I as follows:

  $$OTP = T_i \oplus H(SK),$$

$P_i = A_i \oplus SK \oplus OTP$,
$K'_i = H(ID_i \| URL \| PK \| P`_i \| SS \| CK)$.

Step A3. The server S verifies K`I whether is equal to Ki. If it's not equal, the server rejects the login request. Otherwise, the server proceeds to **Steps 4 – 6** of the authentication phase in Sood et al.'s protocol.

Step A4. If the adversary receives the mutual authentication message $\{M_i, Q_i\}$ from the server, the guessed password is correct. Otherwise, the adversary repeatedly guesses the user's password and executes **Step L1**.

- Denial of Service Attack:

In this attack, an adversary will make the server to cost a large of computation. If the adversary intercepted the user's login request {Ki, CK}, the adversary attacks the server as follows:

Step C1: The adversary by passes all steps of the login phase in Sood et al.'s protocol.

Step C2: The adversary sends the intercepted login request message {Ki, CK} to the server.

Upon receiving the login request message {Ki, CK} from the adversary, the server S authenticates the user as follows:

Step A1. The server S retrieves the user Ui information in the server's database by CK. If it's not CK data in the database, the server rejects the login request. Since the CK is stored in the user's client browser, therefore the server will authenticate the adversary.

Step A2. The server S compared the CUR_TRUST and MIN_TRUST. If CUR_TRUST larger or equal to the MIN_TRUST, the server computes OTP, Pi, and K`I as follows:

$OTP = T_i \oplus H(SK)$,
$P_i = A_i \oplus SK \oplus OTP$,
$K'_i = H(ID_i \| URL \| PK \| P_i \| SS \| CK)$.

Step A3. The server S verifies K`I whether is equal to Ki. If it's not equal, the server rejects the login request. Otherwise, the server proceeds to **Steps 4 – 6** of the authentication phase and compute the common session key SK = H(SS || Pi || Nk || CK || Idi). Since the {Ki, CK} is generated by the legal user Ui, the server will authenticate and send $\{M_i, Q_i\}$ to adversary.

Although the adversary is unable to derive the common session key, he/she will result in costing a large of computation in the server.

4. Conclusion

In this paper, we have shown that the vulnerabilities of Sood et al.'s protocol. Their scheme could not against the on-line guessing password attack and the denial of service attack. The main weakness of Sood et al.'s protocol is that the login request message {Ki, CK} is always the same as that for each login session of the user. To resist these vulnerabilities of Sood et al.'s protocol, concatenate a timestamp Ti to Ki in the login phase: Ki = H(IDi || URL || PK || P`i || SS || CK ||). The user sends the login request message: {Ki, CK, Ti}. The login request message {Ki, CK, Ti} is thus always different from that for each login session of the user.

5. Acknowledgments

This work was partially supported by the Ministry of Science and Technology, Taiwan, under grant MOST 106-3114-E-005-001, MOST 107-2221-E-845-002-MY3, and MOST 107-2221-E-845-001-MY3.

6. References

[1] Hwang M S, Li L H. A new remote user authentication scheme using smart cards [J]. IEEE Transactions on Consumer Electronics, 2000, 46(1): 28-30.

[2] Tsai C S, Lee C C, Hwang M S. Password authentication schemes: Current status and key issues [J]. International Journal of Network Security, 2006, 3: 101-115.

[3] Yang C C, Chang T Y, Hwang M S. The security of the improvement on the methods for protecting password transmission [J]. Informatica, 2003, 14: 551-558.
[4] Zhuang X, Chang C C, Wang Z H, Zhu Y. A simple password authentication scheme based on geometric hashing function [J]. International Journal of Network Security, 2014, 16: 271-277.
[5] Ling C H, Chao W Y, Chen S M, Hwang M S. Cryptanalysis of dynamic identity based on a remote user authentication scheme for a multi-server environment [C]. In: Advances in Engineering Research, 2015, 15: 981-986.
[6] Feng T H, Chao W Y, Hwang M S. Cryptanalysis and Improvement of the Li-Liu-Wu User Authentication Scheme [C]. International Conference on Future Communication Technology and Engineering (FCTE2014), 2014: 103-106.
[7] Chen T Y, Ling C H, Hwang M S. Weaknesses of the Yoon-Kim-Yoo Remote User Authentication Scheme Using Smart Cards [C]. International Conference on Information Technology and Biomedical Engineering, 2014.
[8] He D, Chen J, Hu J. Weaknesses of a remote user password authentication scheme using smart card [J]. International Journal of Network Security, 2011, 13(1): 58-60.
[9] Thandra P K, Rajan J, Murty S A V S. Cryptanalysis of an efficient password authentication scheme [J]. International Journal of Network Security, 2016, 18(2): 362-368.
[10] Pan C S, Tsai C Y, Tsaur S C, Hwang M S. Cryptanalysis of an efficient password authentication scheme [C]. The 2016 3rd IEEE International Conference on Systems and Informatics, 2016: 732-737.
[11] Irawan B, Hwang M S. The weakness of Moon et al.'s password authentication scheme [J]. Journal of Physics: Conference Series, 2018, 1069.
[12] Liao I E, Lee C C, Hwang M S. Security enhancement for a dynamic ID-based remote user authentication scheme [C]. Proceedings of the International Conference on Next Generation Web Services Practices, 2005: 437-440.
[13] Chang T Y, Yang W P, Hwang M S. Simple Authenticated Key Agreement and Protected Password Change Protocol [J]. Computers & Mathematics with Applications, 2005, 49: 703-714.
[14] Guo C, Chang C C, Chang S C. A Secure and Efficient Mutual Authentication and Key Agreement Protocol with Smart Cards for Wireless Communications [J]. International Journal of Network Security, 2018, 20(2): 323-331.
[15] Chiou S Y, Ko W T, Lu E H. A Secure ECC-based Mobile RFID Mutual Authentication Protocol and Its Application [J]. International Journal of Network Security, 2018, 20(2): 396-402.
[16] Ma Y. NFC Communications-based Mutual Authentication Scheme for the Internet of Things [J]. International Journal of Network Security, 2017, 19(4): 631-638.
[17] Hou G, Wang Z. A robust and efficient remote authentication scheme from elliptic curve cryptosystem [J]. International Journal of Network Security, 2017, 19(6): 904-911.
[18] Hwang M S, Yang H W, Yang C Y. An improved hou-wang's user authentication scheme [C]. Lecture Notes in Electrical Engineering, 2018, 514.
[19] Tarek E, Ouda O, Atwan A. Image-based Multimodal Biometric Authentication Using Double Random Phase Encoding [J]. International Journal of Network Security, 2018, 20(6): 1163-1174.
[20] Han L, Xie Q, Liu W. An Improved Biometric Based Authentication Scheme with User Anonymity Using Elliptic Curve Cryptosystem [J]. International Journal of Network Security, 2017, 19(3): 469-478.
[21] Hwang M S, Lee C C, Tang Y L. An Improvement of SPLICE/AS in WIDE Against Guessing Attack [J]. International Journal of Informatica, 2001, 12(2): 297-302.
[22] Amin R. Cryptanalysis and efficient dynamic ID based remote user authentication scheme in multi-server environment using smart card [J]. International Journal of Network Security, 2016, 18(1): 172-181.
[23] Pan H T, Pan C S, Tsaur S C, Hwang M S. Cryptanalysis of efficient dynamic ID based remote user authentication scheme in multi-server environment using smart card [C]. 12th International Conference on Computational Intelligence and Security, 2017: 590-593.

[24] Liu Y, Chang C C, Chang S C. An efficient and secure smart card based password authentication scheme [J]. International Journal of Network Security, 2017, 19(1): 1-10.
[25] Kumar M. A new secure remote user authentication scheme with smart cards [J]. International Journal of Network Security, 2010, 11(2): 88-93.
[26] Kumar M. An enhanced remote user authentication scheme with smart card [J]. International Journal of Network Security, 2010, 10(3): 175-184.
[27] Lee C C, Hwang M S, Yang W P. A flexible remote user authentication scheme using smart cards [J]. ACM Operating Systems Review, 2002, 36(3): 46-52.
[28] Shen J J, Lin C W, Hwang M S. A modified remote user authentication scheme using smart cards [J]. IEEE Transactions on Consumer Electronics, 2003, 49(2): 414-416.
[29] Tang H, Liu X, Jiang L. A robust and efficient timestamp-based remote user authentication scheme with smart card lost attack resistance [J]. International Journal of Network Security, 2013, 15(6): 360-368.
[30] Huang H F, Chang H W, Yu P K. Enhancement of timestamp-based user authentication scheme with smart card [J]. International Journal of Network Security, 2014, 16: 463-467.
[31] Feng T H, Ling C H, Hwang M.S.: An improved timestamp-based user authentication scheme with smart card [C]. The 2nd Congress on Computer Science and Application, 2014: 111-117.
[32] Moon J, Lee D, Jung J, Won D. Improvement of efficient and secure smart card based password authentication scheme [J]. International Journal of Network Security, 2017, 19: 1053-1061.
[33] Chang C C, Lee C Y. A smart card-based authentication scheme using user identity cryptography [J]. International Journal of Network Security, 2013, 16: 139-147.
[34] Sood S K, Sarje A K, Singh K. Inverse Cookie-based Virtual Password Authentication Protocol [J]. International Journal of Network Security, 2016, 13(2): 172-181.