NO SECURITY BY OBSCURITY – WHY TWO FACTOR AUTHENTICATION SHOULD BE BASED ON AN OPEN DESIGN

Jinying Yu and Philipp Brune
Hochschule Neu-Ulm - University of Applied Sciences, Wileystraße 1, D-89231 Neu-Ulm, Germany

Keywords: IT-Security, Identity Management, Two Factor Authentication, Smart Cards.

Abstract: The recently reported security issue possibly compromising the security tokens sold by a major vendor of two factor authentication (2FA) solutions (Schneier, 2011) demonstrates the importance of the basic principle of using an open design for security solutions (Saltzer and Schroeder, 1974). In particular, the safety of such devices should not be based on the use of a secret algorithm or seed value to generate a sequence of one-time passwords (OTP) inside the security token. Instead, we argue in favour of using an open design using pre-generated sequences of OTP that are stored encrypted on the security token. Here, the safety of the solution only relies on the confidentiality of the decryption key and not the design of the solution itself. We illustrate our argumentation by describing a respective authentication scheme and a prototype based on an open design, the latter being used as the basis for the security analysis.

1 INTRODUCTION

There exist several established commercially available solutions for implementing 2FA in an enterprise context. There, the 2FA is achieved by a user's secret knowledge (e.g. a password) in combination with the user's possession of a certain device, the so-called security token (e.g. a smart card or USB device). By means of entering the password, the user obtains a OTP (usually a multi-digit number) from the security token. The user now authenticates with the OTP at the desired service. The advantage of the OTP is that it is only valid for a short period of time, thus providing additional security.

Since Lamport (Lamport, 1981) published a remote user authentication scheme in 1981, many smart card-based 2FA authentication schemes have been proposed in the literature (Yang and Shieh, 1999, Yang et al., 2006 and references therein). Most related publications discuss possible mathematical algorithms to compute the OTP within the security token and the related security problems (Sood et al., 2010, and references therein).

Based on these results, most available 2FA solutions also use an algorithm implemented in parallel inside the security token as well as the authentication server to generate a secret OTP number sequence. Thus, an OTP presented by the user may be verified inside the authentication server by comparing it to the value generated by the implementation inside the server. However, this design embodies two weaknesses:

- Due to inevitable small derivations in the accuracy of the timer clocks inside the security token and the server, the number sequence generation for the OTP runs out of sync after some time. Thus, an additional mechanism for re-synchronization is needed.
- The safety of this approach lies in the unpredictability of the generated number sequence used for the OTP. Thus, the algorithm used for the number generation or at least its seed values need to be kept secret.

Especially the second point prohibits the use of a fully open design for such solutions. The recently reported theft of algorithm and seed value details from a major vendor of 2FA solutions, performed probably by an advanced persistence threat (APT) attack against the companies' servers (Schneier, 2011) in the authors' opinion once again clearly emphasizes the necessity of using an open design for IT security solutions.

For a long time, open design is considered as one of the basic principles of IT security (Saltzer and Schroeder, 1974). Thus, the authors emphasize that it should be applied also to the design of 2FA solutions. Particularly, the safety of the solution

Yu J. and Brune P..
NO SECURITY BY OBSCURITY – WHY TWO FACTOR AUTHENTICATION SHOULD BE BASED ON AN OPEN DESIGN.
DOI: 10.5220/0003610004180421
In *Proceedings of the International Conference on Security and Cryptography* (SECRYPT-2011), pages 418-421
ISBN: 978-989-8425-71-3

should not rely on the obscurity of the OTP sequence generation algorithm or seed value.

One approach for an authentication scheme allowing for an open design would be to abandon the use of a security token in the scheme (Ku, 2004). However, this would be no true 2FA scheme anymore.

Using an open design, a suitable 2FA solution should instead at least fulfil the following requirements:

- No OTP generation inside the security token, thus removing the need for synchroni-sation with the authentication server as well as for keeping the generation algorithm secret.
- No need for a secret seed value or generation algorithm, the full design should be open without a loss of security.
- No need for a special purpose hardware device (e.g. a smart card) used as the security token.
- Thus, neglecting legal issues, a 2FA solution should be at least potentially releasable as open source software (OSS) without a loss of security.

Any solution following these requirements keeps to the basic principle of an open design. In particular, its security would not be based on obscurity.

The rest of the paper is organized as follows: First, an authentication scheme for 2FA using an open design and based on the requirements stated above and a prototype implementation for it are presented. Second, based on the prototype implementation, a security analysis in comparison with the standard smart card-based approach is presented. We conclude with a summary of our findings.

2 AUTHENTICATION SCHEME

Figure 1 shows an overview of a two factor authentication scheme fulfilling the above requirements and used as the reference model in what follows. It contains four main components, namely:

- The *service server* (e.g. a web or virtual private network (VPN) server) the user wants to access,
- The *clients* representing the users and their respective client software to connect to the service server,
- The *authentication server* used to validate the user's credentials and OTP,
- The *storage medium* used as the security token that contains the encrypted OTP sequence.

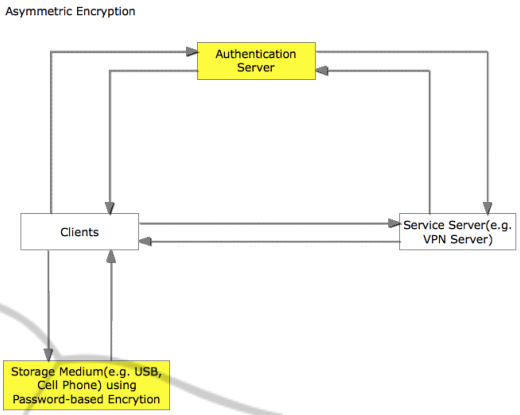

Figure 1: Authentication scheme overview.

Within this scheme, the 2FA is achieved as follows:

In the *registration phase*, a user-specific list of OTP sequence numbers is generated by the authentication server and stored encrypted on the security token and the authentication server. In addition, a client software is installed on the security token that allows to decrypt the OTP list and obtain the next valid OTP by entering a valid PIN or password.

In the *authentication-and-login phase*, a user first authenticates at the (web-based) authentication server with a valid user-id and password (e.g. via HTTPS protocol). The authentication server then presents the user a sequence number indicating which OTP in the list shall be used.

Next, the user enters the OTP sequence number and the password (this password is designed to decrypt the OTP list and is not necessarily the same as the one used to log in at the authentication server) in the client software running on the security token. The software decrypts and displays the corresponding OTP. The user now could authenticate at the service server with user-id and OTP within validity time period of the OTP.

In the *verification phase*, the service server checks the validity of the provided user-id and OTP with the authentication server. Thus, user logs in at the service server with a different OTP each time and each OTP is only valid for a short period of time, e.g. 30 seconds. As the security token any mobile device capable of storing the encrypted OTP list and running the decryption client software may be used.

In order to perform the security analysis, the authors also implemented the described

authentication scheme as a prototype for evaluation and further studies. This implementation uses ubiquitous cell phones as security tokens storing the encrypted OTP list, similar to many commercially available solutions.

The web-based authentication server and the client software components were developed in the the Java programming language, using the Java EE and Java ME frameworks for the server-side and cell-phone components, respectively. The service server calls the authentication server by means of a web service. The OTP list is generated by the authentication server and encrypted using a strong encryption algorithm, e.g. Twofish (Schneier et al., 1998).

Since almost every business person today uses a Java-enabled cell phone, this approach to 2FA a does not require additional equipment for users, thus creating no additional costs. Therefore, also some commercially available 2FA solutions now offer a cell phone-based software replacement of the smart card.

3 SECURITY ANALYSIS

In the security analysis of the prototype implementation for the described authentication scheme, different typical attacks are considered (Lin, 2001).

Since the algorithm for generating the OTP lists is not needed to be defined within the authentication scheme and its implementation needs to reside only on the authentication server and not the security tokens, any known algorithm for generating a sequence of random numbers may be used. It is also possible to easily switch the generation scheme regularly or to use different versions at the same time. Therefore, assuming that always a state-of-the-art algorithm is used, an intruder may not predict or forge the OTP list.

E.g., the HMAC-based OTP algorithm (HOTP) may be used (M'Raihi et al., 2005), which generates OTP values by using keyed hashed message authentication code (HMAC). With the HOTP algorithm, the only thing that has to kept secret is the so-called secret key (not necessarily identical with the user's OTP list decryption key). Any iterative cryptographic hash function could be used within the HOTP algorithm. Since the proposed scheme allows to use arbitrary secret keys and hash functions for the OTP list generation, the difficulty of forgery and prediction attacks is additionally increased. Thus, the authentication scheme is resistant against these attacks as well as an eavesdropping of the sequence number send to the user during the authentication-and-login phase.

If an intruder eavesdrops a valid authentication message during the login process to the service server, the attempt to perform a replay attack will fail in the verification phase since every OTP is invalidated by the authentication server after used once. This is similar for all approaches using an OTP.

However, a major security threat for the scheme is the possible disclosure of the encrypted OTP list from a user's security token. In case the security token (i.e. a cell phone) is stolen or lost, and the user realizes the loss, the stored list may easily be invalidated on the authentication server and replaced by a new one. However, if the user does not realize the loss of the device or only the OTP list is copied from the device, the list may be disclosed by an offline password guessing attack. Although the stored OTP list is encrypted using a strong encrypting algorithm (e.g. Twofish), there is still the danger that an intruder might decrypt it within its period of validity. Depending on the length and the strength of the used password, the encryption might be broken by a password guessing attack. Thus, it is important to use sufficiently strong passwords for the OTP list encryption.

In the future, an interesting solution for this might be the use of self-destructing digital data to prevent copying of the list. E.g., the research project Vanish (Geambasu et al., 2009), conducted by University of Washington, intends to develop such a digital data.

Additionally, the OTP lists should be exchanged frequently. However, this increases the administrative overhead of distributing the OTP lists to the security tokens. Thus, a suitable automation solution for this should be considered.

In summary, any solution implementing the described 2FA scheme, following the stated open design principles and fulfilling the above requirements, in the authors' opinion is not less secure than the frequently used OTP number generation inside the security token.

4 CONCLUSIONS

In conclusion, we presented our opinion on how 2FA should be implemented using an open design approach. We claimed that the recently reported security issue once again demonstrated the

importance of the basic principle of using an open design.

We described a respective two factor authentication scheme and developed a prototype implementation used for the security analysis.

The presented security analysis discusses the resistance of the scheme against the most common attacks. However, further research is needed to prove the resistance of the approach to various other kinds of attacks. In addition, the possibilities of using self destructing data to secure the OTP list need to be further investigated.

REFERENCES

Geambasu, R., Kohno, T., Levy, A., Levy, H. M. (2009). Vanish: Increasing Data Privacy with Self-Destructing Data. In *Proceedings of the USENIX Security Symposium.* Montreal, Canada.

Ku, W. C. (2004). A Hash-Based Strong-Password Authentication Scheme without Using Smart Cards. *SIGOPS Operating Systems Review, 38(1), 29-34.*

Lamport, L. (1981). Password authentication with insecure communication. *Communications of the ACM*, 24(11), 770-772.

Lin, C. L., Sun, H. M., Hwang, T. (2001). Attacks and Solutions on Strong-Password Authentication. *IEICE Transactions on Communications*, E84-B(9), 2622-2627.

M'Raihi, D. Bellare, M., Hoornaert, F., Naccache, D., Ranen, O. (2005). HOTP: An HMAC-Based One-Time Password Algorithm. In *Request for Comments*, 4226. Internet Engineering Task Force. Retrieved May 15, 2011, from http://www.ietf.org/rfc/rfc4226.txt

Saltzer, J. H., Schroeder, M. D. (1974). The Protection of Information in Computer Systems. *Communications of the ACM*, 17(7).

Schneier, B., Kelsey, J., Whiting, D., Wagner, D., Hall, C. (1998). Twofish: A 128-Bit Block Cipher. *AES-Submission.*

Schneier, B. (2011). *Schneier on Security (Blog)*. Retrieved April 8, 2011, from http://www.schneier.com/blog/archives/2011/03/rsa_security_in.html

Sood, S. K., Sarje, A. K., Singh, K. (2010). An Improvement of Xu et al.'s Authentication Scheme using Smart Cards. In COMPUTE '10*, Proceedings of the Third Annual ACM Bangalore Conference. ACM.*

Yang, G., Wong, D., Wang, H., Deng, X. (2006). Formal Analysis and Systematic Construction of Two-Factor Authentication Scheme. In *Information and Communications Security (Lecture Notes in Computer Science)* (pp. 82-91). Berlin / Heidelberg: Springer.

Yang, W. H., Shieh, S. P. (1999). Password Authentication with Smart Cards. *Computers & Security*, 18(8), 727-733.