Pervasive and Mobile Computing 24 (2015) 50–60

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

Efficient provably secure password-based explicit authenticated key agreement

Ou Ruan [a], Neeraj Kumar [b], Debiao He [c], Jong-Hyouk Lee [d,*]

[a] School of Computer Science & Technology, Hubei University of Technology, Wuhan, China
[b] Department of Computer Science and Engineering, Thapar University, Patiala, India
[c] State Key Lab of Software Engineering, School of Computer, Wuhan University, Wuhan, China
[d] Department of Computer Science and Engineering, Sangmyung University, Cheonan, Republic of Korea

ARTICLE INFO

Article history:
Available online 14 June 2015

Keywords:
Key agreement
Explicit authenticated key agreement
Password-based authentication
Provable security
Impersonation attack

ABSTRACT

A password-based authenticated key agreement enables several parties to establish a shared cryptographically strong key over a public unreliable and insecure network using short low-entropy passwords. This authenticated key agreement is definitely required even in Internet of Things (IoT) environments, since no additional device is required. There are only few proposals reported in literature for password-based explicit authenticated key agreement (EAKA). Recently, Zheng et al. proposed a 3-round password-based EAKA protocol. In this paper, we reveal that their protocol is vulnerable to impersonation attack, and the used security definition is not formally treated. We then formalize the security definition of two-party password-based EAKA protocol and improve the construction of Zheng et al. to eliminate its security vulnerabilities. The security of the proposal is formally proved using a new security model.

1. Introduction

The Internet of Things (IoT) is a new emerging technology in which people and things are connected with the ability to process and transfer data over networks. One of crucial requirements for enabling the IoT is to enable secure communications among objects. How to establish reliable and secure communications among parties involved in the communications is still one of the hot topics in modern cryptography and it will continue in near future also. Diffie and Hellman [1] introduced a cryptographic primitive called a key agreement, which is used to provide secure communications over a public unreliable and insecure network with a session key. They considered a passive adversary who merely gathers information, but cannot do anything else. However, this restricted model is far away from the realistic scenario, where a more powerful adversary, e.g., active attacker can do arbitrary actions or automated attacks in IoT environments.

In order to establish a secure session key in such a powerful active adversary model, a variant called authenticated key agreement was proposed. In this scheme, parties should hold some secret information for proving their identity to others. There were two types of secret information: (i) each party holds a private key [2–8] and (ii) parties share a human-memorable password [9–13]. The latter scenario is the case we consider in this paper, and it is definitely the most interesting in practice, since no additional device is required, but just a low-entropy human-memorable password for authenticating the parties. Furthermore, password-based authenticated key agreement (PAKA) requires more works. In this setting, the

* Corresponding author.
E-mail address: jonghyouk@smu.ac.kr (J.-H. Lee).

http://dx.doi.org/10.1016/j.pmcj.2015.06.008
1574-1192/

brute-force on-line dictionary attack is unavoidable, since the authentication means is a short human-memorable password selected from a small possibility. Another off-line guessing attack is more dangerous, where the adversary can check the guessed password against an execution transcript of the protocol off-line. Thus this attack should be avoided in designing PAKA protocols.

An implicit authenticated key agreement (or an authenticated key agreement protocol for simplicity) protocol is an authenticated key agreement protocol without providing key confirmation. Mutual key confirmation, or explicit authentication, means that the party is assured that another party has actually generated the session key, which is an interesting and important additional feature in practice. An authenticated key agreement protocol with mutual key confirmation is called an explicit authenticated key agreement (EAKA) protocol [14]. There are only few proposals for password-based EAKA protocols. Recently, Zheng et al. [9] proposed an efficient 3-round password-based EAKA protocol.

In the paper, we indicate that Zheng et al.'s protocol was vulnerable to impersonation attack and the used security definition was not formally treated. We give a formal security definition of two-party password-based EAKA protocol. Our model is designed for parties to establish a shared cryptographically strong key over a public unreliable and insecure network using short low-entropy passwords under the active adversary. In the proposed model, in order to formalize the adversary's capabilities of faking the mutual authenticators, a new queried oracle called MATest() is introduced. Furthermore, we formalize a mutual authenticated (MA) security definition of the EAKA protocol. Then, in order to overcome the weakness of Zheng et al.'s protocol, we improve their construction and show a formal security proof according to the new security model. Our contributions have three folds. First, we give a formal security definition of two-party password-based EAKA protocol. Second, we improve the construction of Zheng et al. to eliminate its security vulnerabilities. The security of the proposed protocol is formally proved according to the new security model. Third, compared with the construction of Zheng et al., our protocol is more efficient.

The rest of this paper is organized as follows. Section 2 shows related works. Section 3 gives the preliminaries. Section 4 describes the security model of two-party password-based EAKA protocol. Section 5 briefly reviews Zheng et al.'s EAKA protocol. Section 6 presents the proposed two-party password-based EAKA protocol and the detailed analysis of the provable security and the performance of the proposed protocol. Finally, Section 7 concludes this paper.

2. Related works

Bellovin et al. [15] first proposed a PAKA protocol by using an appropriate combination of public-key and secret-key cryptography. Based on Bellovin et al.'s works, many other improved protocols [16–18] were constructed. The security of all these protocols were not formally treated. In 2000, Bellare et al. [19] and MacKenzie et al. [20] demonstrated the provably secure PAKA protocols in the random oracle model. Then many improvements and generalizations in the random oracle model were proposed [21–26]. In contrast, there were only a few PAKA protocols in the standard model. The first PAKA protocol in the standard model was given by Goldreich et al. [27]. Their protocol is impractical because of communication, computation, and round complexity. Nguyen et al. [28] showed some efficiency improvements, but is still far from practical. Katz et al. [29] first designed an efficient PAKA protocol based on the decisional Diffie–Hellman assumption; [30,31] gave some extensions and improvements of [29]. Different constructions of efficient PAKA protocols were given in Refs. [32–34]. Recently, Katz et al. [10] showed a general framework for constructing one-round PAKA protocols in the standard model. These protocols require a common reference string (CRS).

There are few other proposals for password-based EAKA protocols. Bellare et al. [19] proposed a generic transformation that could turn an authenticated key agreement protocol into an EAKA protocol by using the authenticator structure. The hash of the crucial materials and party's ID is used as the authenticator. For example, those protocols [22–24] could be turned into EAKA protocols by this method. However, adding the authenticator structure requires an additional flow, thus these protocols turn out to be at least 4-rounds. Jiang et al. [35] first gave a 3-round password-based EAKA protocol. But their construction was inefficient. Recently, Zheng et al. [9] proposed a very efficient 3-round password-based EAKA protocol. In the paper, we indicate that [9] was vulnerable to impersonation attack, and the used security definition was not formally treated.

3. Preliminaries

The primitives used in the paper are given in this section, including the bilinear pairing, computational Diffie–Hellman assumption, Bilinear Diffie–Hellman assumption, and hash Bilinear Diffie–Hellman assumption. Table 1 shows the notations used throughout this paper.

Definition 1 (*Elliptic Curve Discrete Logarithm Problem (ECDLP)*)**.** Consider $Q_1 = k \cdot Q_2$, where $Q_1, Q_2 \in G$ and $k \in Z_p^*$. The elliptic curve discrete logarithm problem is: given Q_1 and Q_2 in the group, find a number k such that $Q_1 = k \cdot Q_2$.

Definition 2 (*Admissible Pairing [36]*)**.** Suppose G_1 represents an additive cyclic subgroup of G with a large prime order q, G_2 denotes a cyclic multiplicative group with the same order q, and $\hat{e} : G_1 \times G_1 \to G_2$ be a bilinear map. The bilinear map $\hat{e}$ is an admissible pairing if it satisfies the following three properties:

(1) Bilinear: If $Q_1, Q_2 \in G_1$ and $a, b \in_R Z_q^*$, then $\hat{e}(aQ_1, bQ_2) = \hat{e}(Q_1, Q_2)^{ab}$;
(2) Non-degenerate: There exists an element $Q \in G_1$ that satisfies $\hat{e}(Q, Q) = 1$;
(3) Computable: If $Q_1, Q_2 \in G_1$, $\hat{e}(Q_1, Q_2)$ could be computed efficiently.