Attacks on the AKACP Protocol

K. Chalkias[1*], F. Baldimtsi[2], D. Hristu-Varsakelis[1], S. T. Halkidis[1], and G. Stephanides[1]

July, 2010

Abstract

We discuss a recently proposed one-pass authenticated key agreement protocol, by Mohammad, Chen, Hsu and Lo, which was "derived" from their correponding two-pass version and claimed to be secure. We show that this is not the case by demonstrating a number of vulnerabilities.

Keywords: impersonation attacks, key agreement, loss of information, key compromise, denial of service.

1. Introduction

In the recent paper [4] by Mohammad et al. it was shown that the protocols for authenticated key agreement proposed by Elkamchuchi and Eldefrawy in [2] and [3] are vulnerable against key compromise impersonation (KCI) and man-in-the-middle attacks. Mohammad et al. in [4] proposed an alternative two-pass and corresponding one-pass protocols (referred to as AKACP), which they claim withstand the above attacks. The purpose of this note is to show that this claim is incorrect, and to demonstrate several types of serious attacks in those protocols.

[1] Computational Systems and Software Engineering Laboratory, Department of Applied Informatics, University of Macedonia, Greece, chalkias@java.uom.gr, dcv@uom.gr, halkidis@java.uom.gr, steph@uom.gr

[2] Department of Computer Science, Brown University, foteini@cs.brown.edu

* Corresponding author

2. Description of the Two Pass-AKACP Protocol

The AKACP protocol is based on the discrete logarithm problem and consists of three phases: registration, transfer and verification, and key generation. For convenience, the two-pass version of the protocol is depicted in Table 1. For the one-pass version, *B*'s ephemeral key (M_B) is replaced by his public static key (X_B). Thus, *B* does not send anything to *A*, who computes the session key $k_e = X_B^{r_A x_A}$, while *B* computes $k_e = M_A^{x_B}$.

$A(x_A)$	Public information g, X_A, X_B	$B(x_B)$
$r_A, a \xleftarrow{R} \mathbb{Z}_q^*$		$r_B, b \xleftarrow{R} \mathbb{Z}_q^*$
$M_A = X_A^{r_A} = g^{x_A r_A}$, $N_A = g^a$		$M_B = X_B^{r_B} = g^{x_B r_B}$, $N_B = g^b$
$S_A = x_A r_A + x_A + a$		$S_B = x_B r_B + x_B + b$
	$\xrightarrow{(M_A, N_A, S_A)}$	
	$\xleftarrow{(M_B, N_B, S_B)}$	
$S_B^{'} = g^{s_B} = g^{r_B x_B + x_B + b}$		$S_A^{'} = g^{s_A} = g^{r_A x_A + x_A + a}$
$X_B \overset{?}{=} S_B^{'} M_B^{-1} N_B^{-1}$	[verification phase]	$X_A \overset{?}{=} S_A^{'} M_A^{-1} N_A^{-1}$
$X_B \overset{?}{=} g^{r_B x_B + x_B + b} g^{-r_B x_B} g^{-b} = g^{x_B}$		$X_A \overset{?}{=} g^{r_A x_A + x_A + a} g^{-r_A x_A} g^{-a} = g^{x_A}$
$k_e = M_B^{r_A x_A} = g^{r_A r_B x_A x_B}$	[key computation phase]	$k_e = M_A^{r_B x_B} = g^{r_A r_B x_A x_B}$
$k_s = H(k_{e,} M_A, M_B, S_A, S_B, A, B)$		$k_s = H(k_{e,} M_A, M_B, S_A, S_B, A, B)$

Table 1: The Two-Pass AKACP protocol from [4].

3. LoI-Impersonation Attack on One-Pass AKACP

Vulnerability to a loss of information (LoI) attack means that some compromise of other information than private keys, that would not ordinarily be available to an adversary, affects the security of the protocol [1].

Suppose that $x_A, X_A = g^{x_A}$ and $x_B, X_B = g^{x_B}$ are the key pairs for A and B, respectively. We assume that the compromised information is $c = g^{x_A x_B}$. The value of c may be available to an attacker if both a session key k_e and the random value r_A are compromised in any previous run of the one-pass AKACP protocol. We will show that an adversary with knowledge of *c* can impersonate *A* to *B* and vice versa.

The adversary, *E* initiates the one-pass AKACP protocol and computes: $M_A = X_A^{-1}g^t = g^{t-x_A}$, $N_A = g^{k-t}$ and $S_A = k$, where *k* and *t* are random values generated by *E*. Then, *E* sends M_A, N_A, S_A to *B*, and calculates the session key $k_E = c^{-1}X_B^t = g^{-x_A x_B + tx_B}$. *B* checks the equality $X_A = g^{S_A}M_A^{-1}N_A^{-1} = g^k g^{x_A-t} g^{t-k} = g^{x_A}$, which holds; thus, *B* is falsely confident that *E* is *A* and he computes the session key $k_e = M_A^{x_B} = g^{(t-x_A)x_B} = k_E$. It is obvious that if *E* initiates the protocol acting as *B*, the afformentioned attack can similarly be applied to impersonate *B* to *A*.

4. KCI Attack on One-Pass AKACP

If the secret key of an entity *B* is disclosed, then the adversary can obviously impersonate *B*. A key compromise impersonation attack means that the adversary can also impersonate other entities to the victim [1]. Suppose that $x_A, X_A = g^{x_A}$ and $x_B, X_B = g^{x_B}$ are the key pairs of *A* and *B*, respectively. The compromised information is x_B. The adversary, *E*, generates 2 random numbers, *k* and *t*, and computes: $M_A = X_A^{-1} g^t = g^{t-x_A}$, $N_A = g^{k-t}$, and $S_A = k$. Then, *E* sends M_A, N_A, S_A to *B*, and calculates the session key $k_E = (X_A^{-1} g^t)^{x_B} = (g^{-x_A} g^t)^{x_B} = g^{-x_A x_B + t x_B}$.

B checks the equality $X_A = g^{S_A} M_A^{-1} N_A^{-1} = g^k g^{x_A - t} g^{t-k} = g^{x_A}$, which is true; thus, *B* is falsely confident that *E* is *A*. *B* computes the session key $k_e = M_A^{x_B} = g^{(t-x_A)x_B} = k_E$. Clearly, the same attack could also be aimed in a similar way at the two-pass version of the AKACP protocol.

We conclude that the claim by Chalkias et al. [5] still holds; that is, except the CHHSA protocol [6], no other existing one-pass authenticated key agreement (AKA) protocol derived from an original two-pass version can withstand KCI attacks.

5. General Impersonation Attack on One-Pass AKACP

A general impersonation attack means that the attacker can impersonate an entity, say *A*, without knowledge of *A*'s secret key. Suppose $x_A, X_A = g^{x_A}$ and $x_B, X_B = g^{x_B}$ are the key pairs for *A* and *B*, respectively. The adversary, *E*, generates 2 random numbers *k* and *t* and computes $M_A = g^{k-t}$, $N_A = X_A^{-1} g^t = g^{t-x_A}$, and $S_A = k$. Then, *E* sends M_A, N_A, S_A to *B* and calculates the session key $k_E = X_B^{k-t} = g^{(k-t)x_B}$. *B* checks the equality $X_A = g^{S_A} M_A^{-1} N_A^{-1} = g^k g^{t-k} g^{x_A - t} = g^{x_A}$, which is true; thus, *B* is falsely confident that *E* is *A* and he computes the session key $k_e = M_A^{x_B} = g^{(k-t)x_B} = k_E$. This attack is demonstrated in Table 2.

6. General Impersonation Attack on Two-Pass AKACP

Suppose $x_A, X_A = g^{x_A}$ and $x_B, X_B = g^{x_B}$ are the key pairs for *A* and *B*, respectively.

E generates two random numbers, *k* and *t*, and computes $M_A = g^{k-t}$, $N_A = X_A^{-1} g^t = g^{t-x_A}$ and $S_A = k$. Then, *E* sends M_A, N_A, S_A to *B*, receives M_B, N_B, S_B, checks the equality $X_B = g^{S_B} M_B^{-1} N_B^{-1} = g^{x_B}$, and calculates the session key $k_e = M_B^{k-t} = g^{(k-t) r_B x_B}$.

B checks the equality $X_A = g^{S_A} M_A^{-1} N_A^{-1} = g^k g^{t-k} g^{x_A - t} = g^{x_A}$, which is true; thus, *B* is falsely confident that *E* is *A*. *B* computes the session key $k_e = M_A^{r_B x_B} = g^{(k-t) r_B x_B} = k_E$.

E	Public information g, X_A, X_B	$B(x_B)$
$k,t \xleftarrow{R} \mathbb{Z}_q^*$		$r_B, b \xleftarrow{R} \mathbb{Z}_q^*$
$M_E = g^{k-t}$, $N_E = X_A^{-1}g^t = g^{t-x_A}$		$M_B = X_B^{r_B} = g^{x_B r_B}$, $N_B = g^b$
$S_E = k$		$S_B = r_B x_B + x_B + b$
	$\xrightarrow{(M_E, N_E, S_E) \text{ for } ID=A}$	
	$\xleftarrow{(M_B, N_B, S_B)}$	
$S_B^{'} = g^{s_B} = g^{r_B x_B + x_B + b}$		$S_E^{'} = g^{s_E} = g^k$
$X_B \overset{?}{=} S_B^{'} M_B^{-1} N_B^{-1}$	[verification phase]	$X_A \overset{?}{=} S_E^{'} M_E^{-1} N_E^{-1}$
$X_B \overset{?}{=} g^{r_B x_B + x_B + b} g^{-r_B x_B} g^{-b} = g^{x_B}$		$X_A \overset{?}{=} g^k g^{t-k} g^{x_A - t} = g^{x_A}$
$k_e = M_B^{k-t} = g^{(k-t) r_B x_B}$	[key computation phase]	$k_e = M_E^{r_B x_B} = g^{(k-t) r_B x_B}$
$k_s = H(k_{e,} M_E, M_B, S_E, S_B, A, B)$		$k_s = H(k_{e,} M_E, M_B, S_E, S_B, A, B)$

Table 2: Impersonation Attack on Two-pass AKACP protocol [4].

7. User Verification Theorem 4.1 does not hold

According to Theorem 4.1 of [4], only an entity *A*, owning x_A, could pass the verification test $S_A^{'} M_A^{-1} N_A^{-1} = X_A$. We will show that, in fact, the above equation cannot be used as a signature/verification mechanism of *A*. Specifically, we will show how to create a valid signature for *A without* the knowledge of the private key x_A.

The equation under investigation can be rewritten as $KXY = X_A$, where $K = S_A^{'} = g^k$, $X = M_A^{-1}$, $Y = N_A^{-1}$, and $k \in \mathbb{Z}_q^*$ is a random number. We can multiply both sides by $X_A^{-1} = g^{-x_A}$, obtaining $KXYX_A^{-1} = X_A X_A^{-1}$, which equals the group identity element (in $\mathbb{Z}_q^*$ it is 1). We also set $X = K^{-1}$, thus $M_A = K$, and $Y = X_A$, so that $N_A = X_A^{-1}$. Then, the equation in question can be written as $KK^{-1}X_A X_A^{-1} = g^0$, which holds.

Although this argument proves that Theorem 4.1 of [4] does not hold, the above attack can be prevented if one could check and reject the case where $N_A = X_A^{-1}$. However, the attack can then be "extended", by generating one more random number $t \in \mathbb{Z}_q^*$ and using it to obscure matters in N_A by multiplying it by g^t. For a successful unidentifiable attack on the signature mechanism of [4], we set $S_A = k$, $M_A = g^{k-t}$ and $N_A = X_A^{-1}g^t$. Using the aforementioned settings for S_A, M_A, N_A, the verification test is passed, because $S_A^{'} = g^k g^{t-k} g^{x_A - t} = g^{x_A}$.

7.1 A change-ID Man-in-the-Middle Attack

Due to the flaw found in Theorem 4.1 of [4], an active change-ID man-in-the-middle attack is possible. In fact, this attack is a subset of unknown key-share (UKS) attacks but not as strong as a general UKS. More precisely, a change-ID attacker arranges things so that a false key confirmation (or user authentication) is possible, misleading the victim to believe that he/she successfully communicates with an other entity. This attack is just "a step down" from a general UKS attack, meaning that the key confirmation stage has passed successfully, but the two parties cannot communicate. Typically, such an attack is used for denial of service (DoS), as key confirmation will always be possible but users will not be able to communicate. In the following, we describe a method for change-ID man-in-the-middle attacks, where *E* arranges matters to pass the verification phase as being the *C* entity, without knowing *C*'s private key.

Suppose that *A* wants to share a key with *B* and starts the AKACP protocol (both one pass and two pass versions are vulnerable). *E* intercepts the data sent from *A* to *B* and forwards a new N_A, $N_A^* = N_A M_A X_C^{-1} = g^{a+x_A-x_C}$. Upon receiving S_A, M_A, N_A^*, *B* executes the verification $S_A^/ M_A^{-1} N_A^{*-1} = g^{r_A x_A + x_A + a} g^{-r_A x_A} g^{-a-x_A+x_C} = g^{x_C} = X_C$ and accepts it, believing that he has established a key with *C*. In a real life scenario where *B* might be a service provider, this attack would work as an active DoS, leading to continuous fake authentications.

It is noted that the above attack does not lead to a general UKS attack. In the AKACP protocols it is easy to establish a key, k_e, through communication with an attacker. However, the inclusion of the identities' strings in the hash input saves the protocol from accepting a key from a dishonest entity. This is a property that holds in general in key establishment protocols where the key is calculated as the hash product of the computed secret information (such as k_e in [4]) together with the identities of the communicating parties. Thus, although there are also other ways to protect against UKS attacks, the inclusion of the identities of the parties in the computed hashed session key is a simple and fast method to avoid such kinds of threats.

References

[1] *Key Agreement Protocols and their Security Analysis*, Blake-Wilson S., Johnson D. and Menezes A., Sixth IMA International Conference on Cryptography and Coding, Cirencester, England, 17-19 December 1997.

[2] *A New Approach for Key Controlled Agreement*, Elkamchuchi H. and Eldefrawy M., 24 National Radio Science Conference-NRSC2007, Ain Shams University, Egypt, pp. 1-7, Mar. 2007.

[3] *An Efficient and Confirmed Protocol for Authentication Key Agreement* Elkamchuchi H. and Eldefrawy M., 25 National Radio Science Conference-NRSC2008, Tanta University, Egypt, pp. 1-8, Mar. 2008.

[4] *Cryptanalysis and Enhancement of Two-pass Authenticated Key Agreement with Key Confirmation Protocols*, Mohammad Z., Chen Y.-C., Hsu C.-L. and Lo C.C., IETE Technical Review Vol. 27, Issue 3, May-June 2010.

[5] *Two Types of Key-Compromise Impersonation Attacks against One-Pass Key Establishment Protocols* Chalkias K., Baldimtsi F., Hristu-Varsakelis D., and Stephanides G., vol. 23, 4th International Conference ICETE 2007, Springer, pp. 227-38, 2007.

[6] *A Provably Secure One-Pass Two-Party Key Establishment Protocol* Chalkias K., Halkidis S.T., Hristu-Varsakelis D., Stefanides G., vol. 4990, 3rd International SKLOIS Conference on Information Security and Cryptology - Inscrypt 2007, Springer-Verlag, pp. 108-22.