# Cryptanalysis of an ID-based Password Authentication Scheme using Smart Cards and Fingerprints

Michael Scott

School of Computer Applications
Dublin City University
Ballymun, Dublin 9, Ireland.
mike@computing.dcu.ie

**Abstract.** In a paper recently published in the ACM Operating Systems Review, Kim, Lee and Yoo [1] describe two ID-based password authentication schemes for logging onto a remote network server using smart cards, passwords and fingerprints. Various claims are made regarding the security of the schemes, but no proof is offered. Here we show how a passive eavesdropper, without access to any smart card, password or fingerprint, and after passively eavesdropping only one legitimate log-on, can subsequently log-on to the server claiming any identity.

## 1 Introduction

The problem being addressed in [1] (available online at `http://140.134.25.71/~education/group-meeting/040106_2.pdf`) is that of securely logging on to a remote network server. In this context entity authentication is clearly an important issue. Traditionally there are three methods for authentication based on who you are (using biometrics, for example using a fingerprint), what you know (a password or pass-phrase) and what you possess (an uncloneable token, for example a smart-card). It is of interest to devise a scheme that uses all three methods for maximum security

In [1] such a scheme is suggested, with the claimed additional advantage that the user gets to choose and, if they wish, subsequently change their password. However as we shall see the suggested scheme is completely insecure, even against a passive eavesdropper who does not possess any smart-card or password, and irrespective of their fingerprints. In fact two protocols are suggested. One uses a timestamp to prevent replay attacks, and the second uses a nonce for the same purpose. We describe our attack on the nonce-based scheme, although it also applies to the timestamp version.

## 2 The Kim-Lee-Yoo protocol

We use the same notation as [1]. Their protocol, which we have simplified a little for clarity, consists of 3 phases. These are registration, login and verification.

There are three public parameters $n$, $g$ and $f$, where $n$ is a large prime, $g$ is a generator of order $n-1$, and $f$ is a one-way function. $U_i$ denotes each legal user of identity $ID_i$, whose chosen password is $PW_i$. The server has a master secret $SK$.

To register the user decides on their password and submits $ID_i$ and $PW_i$ over a secure channel (perhaps face-to-face) to the registration authority associated with the server. When satisfied that $U_i$ is indeed who they claim to be, a smart-card identifier $CID_i$ is generated, and a smart card is issued to $U_i$ containing $n$, $g$, $f$, $ID_i$, $CID_i$, $S_i$ and $h_i$, where

$$\begin{aligned} S_i &= {ID_i}^{SK} \pmod n \\ h_i &= g^{PW_i.SK} \pmod n \end{aligned}$$

Note that the fingerprint is used to establish ownership of the smart-card. It is also used as a source of random numbers.

The login phase consists of four steps, carried out remotely over an insecure channel. First the smart-card for user $U_i$ sends a login request consisting of $\{ID_i, CID_i\}$ to the remote server. The server verifies the validity of $ID_i$ and $CID_i$, generates a random $r_s$ and creates a nonce

$$N = f(CID_i, r_s)$$

which is sent back to the smart-card. The smart card then generates a random $r_i$ and computes

$$\begin{aligned} X_i &= g^{r_i.PW_i} \pmod n \\ Y_i &= S_i.{h_i}^{r_i.N} \pmod n \end{aligned}$$

Finally the smart-card sends $X_i$ and $Y_i$ to the server. This completes the login phase.

In the verification phase the server decides whether or not to permit a login for this user. This is done by checking if the following equation holds

$${Y_i}^{SK^{-1}} \equiv ID_i.{X_i}^N \pmod n$$

## 3 The Cryptanalysis

Clearly to break this scheme, that is to login without possession of the smart-card or the password (or indeed the fingerprint), it is sufficient to come up with values for $X_i$ and $Y_i$ which pass the verification test. By substituting for $S_i$, $h_i$ and $X_i$ it is not difficult to see that

$$Y_i = (ID_i.{X_i}^N)^{SK} \pmod n$$

An attacker does not know $SK$, but by eavesdropping a legitimate login by user $U_j$, an attacker can calculate values

$$G = ID_j.{X_j}^N \pmod{n}$$

from transmitted values, and find

$$G^{SK} = Y_j \pmod{n}$$

As it turns out possession of this pair $\{G, G^{SK}\}$ is as good as knowing $SK$ and sufficient to allow an attacker to login claiming any identity. To login as user $U_i$ start the login procedure by submitting eavesdropped values for $\{ID_i, CID_i\}$. When the server sends back the nonce $N$ the attacker responds by submitting

$$X_i = G/{ID_i}^{N^{-1}} \pmod{n}$$
$$Y_i = (G^{SK})^N \pmod{n}$$

It is easy to see that this pair will satisfy the verification procedure, and so the attacker successfully logs in. To prevent detection of this particular attack, the pair $\{G^r, (G^{SK})^r\}$ which is equivalent to $\{(G^r), (G^r)^{SK}\}$ can be used in place of $\{G, G^{SK}\}$, where $r$ is any random number.

## 4 Conclusions

The schemes described in [1] are completely insecure. Do not use them.

## References

1. H. S. Kim, S. W. Lee and K. Y. Yoo. "ID-based Password Authentication Scheme using Smart Cards and Fingerprints", ACM Operating Systems Review, Vol. 37,No. 4, pp. 32–41,October 2003.