On the Security of a Proxy Blind Signature Scheme over Braid Groups

Manoj Kumar
Department of Mathematics
Rashtriya Kishan Postgraduate College
Meerut- Karnal Road Shamli-Muzaffarnagar, Utter Pradesh-India- 247776
E-mail: yamu_balyan@yahoo.co.in

Abstract

A proxy blind signature scheme is the combination of proxy signature and blind signature scheme. In 2009,Verma proposed a proxy blind signature scheme over braid groups. Verma claimed that the proposed scheme is secure against all possible security lapses and also satisfy all essential security attributes.This paper analyzes Verma's proposed scheme and found that this scheme suffers with the serious security vulnerabilities. This paper show that the proposed scheme does not satisfy unforgeability and unlinkability, which are two essential security requirement of a secure proxy blind signature scheme.

1. Introduction

In 1984 [4] Chaum introduced the concept of blind signature scheme. A blind signature scheme is a protocol played by two parties in which a user can obtain the signature of a valid signer on a desired message and the signer learns nothing about the message. With such properties, the blind signature schemes are useful in several applications such as e-voting and e-payment [5]. For more detail on blind signature schemes, please refer to [4, 14, 20, 21, 25, 28, 37, 38]

On the other side, in 1996, Mambo Usuda and Okamoto [29] introduced the concept of proxy signature scheme. The proxy signature scheme allowed an entity called original signer to delegate its signing capabilities to another entity called proxy signer and the proxy signer signs message on behalf of the original signer. Once the signature verifier receives the proxy signature, she /he can check the validity of the signature and identify the proxy signer and also verifies the original signer's agreement on the signed message. Based on delegation type, Mambo et al. [29] classified proxy signatures as

- Full Delegation
- Partial Delegation
- Delegation by Warrant

In the full delegation, the original signer gives his secret key to the proxy signer. In the partial delegation, the original signer generates a proxy signature key by using his secret key and then transfers this key to the proxy signer, who uses the proxy key to sign the message on behalf of original signer. In the delegation by warrant, the proxy signer first obtains the warrant, which is a certificate composed of a message part and a public signature key from the original signer, and then uses the corresponding secret key to sign the concern message. The resulting signature consists of the created signature and the warrant. For more detail on proxy signature schemes, please refer to [1, 2, 3, 8, 16, 24, 26, 27, 29, 30, 31, 32, 33, 34, 35, 36, 37].

A proxy blind signature scheme is a digital signature scheme which combines the properties of proxy signature with blind signature schemes. In a proxy blind signature scheme, the proxy signer is allowed to generate a blind signature on behalf of the original signer. For more detail, please refer to [7, 14, 15]. A proxy signature scheme should satisfy the following basic security requirements

1.1 Security Requirements

The security requirements for a secure proxy blind signature scheme are specified in [2]. which are explained below.

- **SR**$_1$. VERIFIABILITY: From the proxy signature, any verifier can be convinced of the original signer's agreement on the signed message.
- **SR**$_2$.STRONG UNFORGEABILITY:PROXY PROTECTION: A valid proxy signature can only be generated by proxy signer. This means that valid proxy signature cannot be created by the original signer or any third party who is not designated as proxy signer. In other

words, we can say that only the delegated proxy signer can generate valid partial proxy signature. Even the original signer cannot masquerade as a proxy signer.

- **SR**$_3$.STRONG IDENTIFIABILITY: Anyone can determine the identity of the corresponding proxy signer form a proxy signature.
- **SR**$_4$.STRONG UNDENIABILITY (NONREPUDIATION): Any valid proxy signature must be generated by proxy signer. Therefore, proxy signer can not deny that he/she has signed the message. In addition, the original signer cannot deny having delegated the power of signing messages to the proxy signer.
- **SR**$_5$.DISTINGUISHABILITY: The verifier can distinguish the original and proxy signature efficiently.
- **SR**$_6$.SECRET KEY DEPENDENCIES: Proxy signature key or the delegation information can be computed only with the help of original signer's secret key.
- **SR**$_6$.TIME CONSTRAINT: The proxy signing key can be used only during the delegated period. Once the proxy key expire, the proxy signature generated by using this key become invalid.
- **SR**$_7$.PREVENTION OF MISUSE : The proxy signer is restricted to transfer the proxy key to someone else. The proxy signer also can not use proxy key for purposes other than generating a valid proxy signature. In case of misuse, the responsibility of the proxy signer should be determined from the warrant.
- **SR**$_8$.UNLINKABILITY:When the signature is revealed, the proxy signer can not identify the association between the message and the blind signature he generated.

In 2000, Ko et. al. proposed a key agreement protocol and a public key encryption scheme [23] based upon braid groups [9, 10, 11, 12]. The schemes based upon braid groups [15, 17, 18, 19, 19, 22, 23] are analogous to the Diffie-Hellman key agreement scheme and the ElGamal encryption scheme on abelian groups. Their basic mathematical problem is the Conjugacy Problem (CP) on braids: For a braid group B_n, we are asked to find a braid *a* from $u, b \in B_n$satisfying $b = aua^{-1} \in B_n$. The security is based on the *Diffie-Hellman Conjugacy Problem* (**DHCP**) to find $baua^{-1}b^{-1} \in B_n$ for given $u, aua^{-1}, bub^{-1} \in B_n$for *a* and *b* in two commuting subgroups of B_n respectively. In 2008, Verma introduced a proxy blind signature scheme over braid groups [15]. Verma's proposed scheme [15] is based on the conjugacy search problem over braid groups. Verma claimed that their proposed proxy blind signature scheme is partial protected proxy signature. Verma [15]also proved that the all security parameters are satisfied by our scheme.This paper analyzes Verma's proposed scheme and found that this scheme suffers with the security flaws. This paper is organized as follows.
Section 2 provides a brief idea of braid group and explain the difficulty of the computational version. In section 3, we review Verma's proxy blind signature scheme over braid group. The securities flaws of Verma's proposed scheme are discussed in section 4. Finally, we conclude the work in section 5.

2 Braid Group and Conjugacy problem

In this section, we give the basic definitions of braid groups and discuss some hard problems on those groups. For more information on braid groups, word problem and conjugacy problem,refer to the papers [6, 9, 10, 11, 12, 15, 17, 18, 19, 22, 23]. A braid is obtained by laying down a number of parallel strands and intertwining them so that they run in the same direction. For each integer $n \geq 2$, the *n*-braid group B_n is the group generated by $\sigma_1\sigma_2, ..., \sigma_{n-1}$ with the relations $\sigma_i\sigma_j = \sigma_j\sigma_i$ where $|i - j| \geq$2 and $\sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1}$ otherwise. The number *n* is called the braid index and each element of B_n is called *n* - braid. Two braids *x* and *y* are said to be conjugate if there exist a braid *a* such that $y = axa^{-1}$. For $m < n$,B_m can be considered as a subgroup of B_n generated by $\sigma_1\sigma_2, ..., \sigma_{m-1}$.

In Braid Cryptography, let *G* be a non-abelian group and $u, a, b, c \in G$. In order to perform the Diffie- Hellman key agreement on *G*, we need to choose *a, b* in *G* satisfying *ab* = *ba* in the *DHCP*(Diffie-Hellman Conjugacy Problem). Hence, we introduce two commuting subgroups $G_1, G_2 \subset G$ satisfying ab = ba for any $a \in G_1$ and $b_2 \in G_2$. More precisely, the the braid cryptography are based on the following decision problems.

- `Input`:
  A non-abelian group *G*, two commuting subgroups $G_1, G_2 \subset G$
- `Conjugacy Problem`:
  Given (u, aua^{-1}) with $u, a \in G$, compute *a*. (Note that if we denote aua^{-1} by u^a, it looks like the DLP.)
- `Diffie-Hellman Conjugacy Problem`:
  Given(u, aua^{-1}, bub^{-1}) with $u \in G, a \in G_1$ and $b \in G_2$, compute $baua^{-1}b^{-1}$.
- `Decisional Diffie-Hellman Conjugacy Problem`:
  Given$(u, aua^{-1}, bub^{-1}, cuc^{-1})$ with $u, c \in G, a \in G_1$ and $b \in G_2$, decide whether *c* = *ba*.

In braids, we can easily take two commuting subgroups G_1 and G_2 of B_n(For simplicity, we only consider a braid group with an even braid index. But it is easy to extend this to an odd braid index.). For example, $G_1 = LB_n$ (resp. $G_2 = RB_n$) is the subgroup of B_n consisting of braids made by braiding left $n/2$ strands(resp. right $n/2$ strands) among n strands. Thus LB_n is generated by$\sigma_1\sigma_2, ..., \sigma_{n/2-1}$ and RB_n is generated by $\sigma_{n/2-1}, ..., \sigma_{n-1}$. Then we have the commutative property that for any $a \in G_1$and $b \in G_2$, *ab* = *ba*. We choose a sufficiently complicated $(l + r)$-braid $\alpha \in B_{l+r}$.Then following is a one-way function.

$$f : G_1 \times G_n \longrightarrow G_n \times G_n, f(a, x) = (axa^{-1}, x).$$

There is an efficient time algorithm [17] for a given pair (a, x) to compute axa^{-1}, but all the known attacks need exponential time to compute a from (axa^{-1}, x). This one-way function is based on the difficulty of conjugacy problem.

3 Review of Verma's Purposed Scheme

This section reviews Verma's proxy blind signature scheme over braid group [15]. In this scheme, to sign a message $m \in [0, 1]^*$, the original signer Alice delegates his signing capability to a proxy signer Bob.

3.1 Key Generation

Each user *u* does the following steps.

- Selects a braid $x_u \in_R B_n$ such that $x_u \in SSS(x_u)$.
- Choose $x_u, a_u \in_R RSSBG(x_u, d)$.
- Return public key as $pk = (x_u, x_u^{'})$ and secret key $sk = a$.

3.2 Proxy Key Generation

Bob gets the proxy key pair as follows.

- The original signer Alice selects a braid $\alpha_o \in_R B_n$.
- Alice computes $t_o = a_o x a_o^{-1}$. Then, she sends the pair (α_o, t_o) to Bob through a secure channel.
- Bob checks whether $t_o x_o^{'} \sim \alpha_o x_o$. If it is hold, he accept the key, otherwise reject it.

3.3 Proxy Blind Signature Generation

When the proxy signer Bob signs a document on behalf of original signer Alice, he computes the following steps.

- Proxy signer chooses $b \in_R B_n(l)$ and computes $\alpha = bx_p b^{-1}$ and sends (t_0, α) to the user.
- `Blinding`: User chooses $\delta \in_R B_n(l)$ and computes $t_0^{'} = \delta t_0 \delta^{-1}, \alpha^{'} = \delta\alpha\delta^{-1}$ and $h = H(H_1(t_0^{'} x_0^{'}||m)$ and sends *h* to the proxy signer.
- Proxy signer computes $\beta = bhb^{-1}, \gamma = ba_p^{-1} h a_p b^{-1}$ and sends (β, γ) to the user .
- `Unblinding`: User computes $\beta^{'} = \delta\beta\delta^{-1}, \gamma^{'} = \delta\gamma\delta^{-1}$ and display $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_0^{'})$ as a proxy blind signature on message *m*.

3.4 Proxy Blind Signature verification

To verify the proxy blind signature on a message *m*, a verifier computes the following steps.

- Verifier computes $h = H(H_1(t_o^{'} x_o^{'})\|m)$.
- Verifier checks whether $\alpha^{'} \sim x_p, \beta^{'} \sim h, \gamma^{'} \sim h, \alpha^{'}\beta^{'} \sim x_p h, \alpha^{'}\gamma^{'} \sim x_p^{'} h$, if it is hold,accept the signature,otherwise reject it.

4 Security Analysis of Verma's Proxy Blind Signature Scheme

This section analyzes the security of a proxy blind signature scheme over braid group[15]. According to Verma[15], the proposed scheme satisfies all the security requirements strongly. Moreover, Verma claimed that there is no effect of the revelation of the delegation pair on the security of the proposed proxy blind signature scheme over braid groups. we feel that these claims are not true. In Verma's scheme,the original signer sends the signing key in the form of a pair (α_o, t_o)to Bob through a secure channel. The proxy singer verify the validity of this pair by checking the congruence $t_o x_o^{'} \sim \alpha_o x_o$. If it is hold, he accepts(α_o, t_o)as a proxy secret key to sign a message *m*. It is very interesting to see that in the proxy key generation, the value t_o is kept secret, while the value $t_o^{'}$ is transmitted to the third party in the proxy blind signature generation phase.The braid t_o and $t_o^{'}$ are conjugates. The conjugacy relation between t_o and $t_o^{'}$ helps an attacker to mounts an attack on Verma's proposed proxy blind signature scheme. On this statement, we successfully listing several interesting forgery attacks on Verma's scheme [15] . The following section presents a security analysis of the Verma's proposed proxy blind signature scheme over braid group[15] in detail.

4.1 Balyan's Cloning Attack

In this section, we introduced a different kind of security attack on Verma's proposed proxy blind signature scheme over braid group[15]. We named our attack as Balyan's Cloning Attack or simply Cloning Attack. Cloning attack means an antagonist can generate a valid proxy blind signature (Cloned Proxy Blind Signature) only with the help of a previously generated proxy blind signature. The interesting fact is that Cloned Proxy Blind Signature can be generated without any knowledge of proxy secret key or other related secret parameters. The Cloned Proxy Blind Signature looks like as original signature and also satisfy all the properties/ requirements of the original signature. The following steps show that how an antagonist can mount a cloning attack on Verma' s proxy blind signature over braid group. Suppose the attacker had a valid blind signature $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_o^{'})$ on the message *m* generated by a valid proxy signer. Now, an antagonist, Charlie can generate a Cloned Proxy Blind Signature $(\alpha^{''}, \beta^{''}, \gamma^{''}, t_o^{'})$ as follows.

- Charlie selects a braid $\xi \in_R B_n$.
- Charlie computes $\alpha^{''} = \xi\alpha^{'}\xi^{-1}$.
- Charlie computes $\beta^{''} = \xi\beta^{'}\xi^{-1}$.
- Charlie computes $\gamma^{''} = \xi\gamma^{'}\xi^{-1}$.

Now, we show that the fabricated proxy blind signature $(\alpha^{''}, \beta^{''}, \gamma^{''}, t_o^{'})$ is a Cloned Proxy Blind Signature on the message *m*.

4.1.1 Cloned Proxy Blind Signature Verification

To check the validity of a Cloned Proxy Blind Signature, a verifier runs the following steps.

- Verifier computes $h = H(H_1(t_o^{'}x_o^{'})\|m)$.
- Verifier checks whether $\alpha^{''} \sim x_p, \beta^{''} \sim h, \gamma^{''} \sim h, \alpha^{''}\beta^{''} \sim x_p h, \alpha^{''}\gamma^{''} \sim x_p^{'}h$, if it is hold,accept the signature,otherwise reject it.

Since, in the verification phase. the first step is same as the the original scheme, therefore this always holds truly. Obviously, all the conjugacy relations $\alpha^{''} \sim x_p, \beta^{''} \sim h, \gamma^{''} \sim h, \alpha^{''}\beta^{''} \sim x_p h, \alpha^{''}\gamma^{''} \sim x_p^{'}h$, will hold truly. It can be proved easily as follows.

$$\begin{aligned}
\alpha^{''} &= \xi\alpha^{'}\xi^{-1} \\
&= \xi\delta\alpha\delta^{-1}\xi^{-1} \\
&= \xi\delta b x_p b^{-1}\delta^{-1}\xi^{-1} \\
&= \xi\delta b x_p(\xi\delta b)^{-1} \Rightarrow \alpha^{''} \sim x_p.
\end{aligned}$$

$$\begin{aligned}
\beta^{''} &= \xi\beta^{'}\xi^{-1} \\
&= \xi\delta\beta\delta^{-1}\xi^{-1} \\
&= \xi\delta b h b^{-1}\delta^{-1}\xi^{-1} \\
&= \xi\delta b h(\xi\delta b)^{-1} \Rightarrow \beta^{''} \sim h.
\end{aligned}$$

$$\begin{aligned}
\gamma^{''} &= \xi\gamma^{'}\xi^{-1} \\
&= \xi\delta\gamma\delta^{-1}\xi^{-1} \\
&= \xi\delta b a_p^{-1} h a_p b^{-1}\delta^{-1}\xi^{-1} \\
&= \xi\delta b a_p^{-1} h(\xi\delta b a_p^{-1})^{-1} \Rightarrow \gamma^{''} \sim h.
\end{aligned}$$

$$\begin{aligned}
\alpha^{''}\beta^{''} &= [\xi\delta b x_p(\xi\delta b)^{-1}][\xi\delta b h(\xi\delta b)^{-1}] \\
&= [\xi\delta b x_p(b^{-1}\delta^{-1}\xi^{-1})][\xi\delta b h(\xi\delta b)^{-1}] \\
&= \xi\delta b x_p b^{-1}\delta^{-1}\xi^{-1}\xi\delta b h(\xi\delta b)^{-1} \\
&= \xi\delta b x_p h(\xi\delta b)^{-1} \Rightarrow \alpha^{''}\beta^{''} \sim x_p h.
\end{aligned}$$

$$\begin{aligned}
\alpha^{''}\gamma^{''} &= [\xi\delta b x_p(\xi\delta b)^{-1}][\xi\delta b a_p^{-1} h(\xi\delta b a_p^{-1})^{-1}] \\
&= [\xi\delta b x_p(b^{-1}\delta^{-1}\xi^{-1})][\xi\delta b a_p^{-1} h a_p b^{-1}\delta\xi^{-1}] \\
&= \xi\delta b x_p(b^{-1}\delta^{-1}\xi^{-1})\xi\delta b a_p^{-1} h a_p b^{-1}\delta\xi^{-1} \\
&= \xi\delta b a_p^{-1} a_p x_p a_p^{-1} h a_p b^{-1}\delta\xi^{-1} \\
&= \xi\delta b a_p^{-1}(a_p x_p a_p^{-1}) h a_p b^{-1}\delta\xi^{-1} \\
&= \xi\delta b a_p^{-1} x_p^{'} h a_p b^{-1}\delta\xi^{-1} \Rightarrow \alpha^{''}\gamma^{''} \sim x_p^{'}h.
\end{aligned}$$

It is also clear that original proxy blind signature $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_o^{'})$ and Cloned Proxy Blind Signature $(\alpha^{''}, \beta^{''}, \gamma^{''}, t_o^{'})$ are statistically indistinguishable. Since,the Cloned Proxy Blind Signature $(\alpha^{''}, \beta^{''}, \gamma^{''}, t_o^{'})$ also satisfies all the verification steps successfully, therefore the verifier accept the Cloned Proxy Blind Signature as a real proxy blind signature.

4.2 Misuse of delegation Power

In Verma's scheme[15], the delegation pair includes neither the identity information of the proxy signer nor the limit on delegated messages. In the proposed scheme [15], the proxy signer can further delegate the proxy key to someone else who can also perform the signing operation on behalf of original signer.In this way,a third party has same signing capability as a designated proxy signer. Furthermore,the delegation pair does not contain any information about the duration period of delegated power. It means the proxy signer has been selected permanently. Once a proxy signer selected,then he will remain the proxy signer forever. The original signer's delegation power does not contain any information about the qualification of the messages on which the proxy signer can sign. The proxy signer can select any message of his choice and then sign it. In these ways,the proxy signer is able to misuse his delegated signing capabilities and the original signer can not restrict the proxy signer for misuse her delegation power. Consequently, in Verma's scheme [15], the essential security requirements are not satisfied.

4.3 Original signer changing attack

In Verma's proposed scheme [15]there is a need of secure channel to deliver the delegation information. Verma claimed that his scheme is still secure even if an attacker intercept the delegation pair. We observe that his claim is not true. The following steps prove that how an antagonist can mount an original signer attack on Verma' s scheme by the interception of the delegation pair.

4.3.1 Generation of Fabricated Proxy Key

- The antagonist Charlie intercept the delegation pair(α_o, t_o).
- The antagonist selects a braid $\alpha_c \in_R B_n$ and computes $t_c = a_c x a_c^{-1}$. Then, she replaces the pair(α_o, t_o)with (α_c, t_c) and sends this pair to the proxy singer Bob.
- Bob checks whether $t_c x_c^{'} \sim \alpha_c x_c$. If it is hold, he accept the key, otherwise reject it. Obviously, This conjugacy relation will hold truly. It can be seen easily that the delegation pairs (α_o, t_o) and (α_c, t_c) are statistically indistinguishable. Since,there is also no information about the original signer in the delegation pair(α_o, t_o) strictly, therefore the Bob can not determine the identity of the original signer explicitly.

Now in place of delegation pair (α_o, t_o),the proxy signer uses the fabricated delegation pair (α_c, t_c). It can be seen easily that this replacement does not effect the proxy blind signature generation and verification phases.

4.4 Proxy Signer Changing Attack

In Verma's proposed scheme [15] the delegation pair does not include the identity of the proxy signer. In this situation, an interesting attack can be mounted on Verma's proposed scheme [15]. In this attack, an antagonist can become the proxy signer in place of a valid proxy signer Bob. we call this attack proxy signer changing attack. in this attack, any antagonist Charlie can generate a valid proxy blind signature on a message m_u of user's choice. Verma claimed that his scheme is still secure even if an attacker intercept the delegation pair. Again, We prove that his claim is not true.The following steps prove that how an antagonist can mount proxy signer changing attack on Verma' s scheme [15] by the interception of the delegation pair.

4.5 Generation of Fabricated Proxy Blind Signature

- The antagonist Charlie intercept the delegation pair(α_o, t_o).
- Charlie chooses $b_c \in_R B_n(l)$ and computes $\alpha_c = b_c x_c b_c^{-1}$ and sends (t_0, α_c) to the user.
- `Blinding`: User chooses $\delta \in_R B_n(l)$, computes $t_0^{'} = \delta t_0 \delta^{-1}, \alpha^{'} = \delta \alpha_c \delta^{-1}$, $h = H(H_1(t_0^{'} x_0^{'} || m_u)$ and returns *h* to the signer.
- Charlie computes $\beta_c = b_c h b_c^{-1}, \gamma_c = b_c a_c^{-1} h a_c b_c^{-1}$ and returns (β_c, γ_c) to the user.
- `Unblinding`: User computes $\beta^{'} = \delta \beta_c \delta^{-1}, \gamma^{'} = \delta \gamma_c \delta^{-1}$ and display $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_0^{'})$ as a proxy blind signature on a message m_u

4.5.1 Verification of Fabricated Proxy Blind Signature

To verify the Fabricated proxy blind signature, a verifier computes the following steps.

- Verifier computes $h_c = H(H_1(t_o x_o^{'}) \| m_u)$.
- Verifier checks whether $\alpha^{'} \sim x_c, \beta^{'} \sim h, \gamma^{'} \sim h, \alpha^{'} \beta^{'} \sim x_c h, \alpha^{'} \gamma^{'} \sim x_c^{'} h$, if it is hold,accept the signature,otherwise reject it.

Obviously, all the conjugacy relations will hold truly. It can be seen easily that the fabricated proxy blind signature $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_0^{'})$and original proxy blind signature are statistically indistinguishable. Since,neither the delegation pair(α_o, t_o) nor the proxy signature provide any information about the proxy signer strictly, therefore the third party can not determine the identity of the proxy signer explicitly. Thus , the verifier accept the fabricated proxy blind signature as proxy blind signature.

- Proxy signer chooses $b \in_R B_n(l)$ and computes $\alpha = b x_p b^{-1}$ and sends (t_0, α) to the user.
- `Blinding`: User chooses $\delta \in_R B_n(l)$ and computes $t_0^{'} = \delta t_0 \delta^{-1}, \alpha^{'} = \delta \alpha \delta^{-1}$ and $h = H(H_1(t_0^{'} x_0^{'} || m)$ and sends *h* to the proxy signer.
- Proxy signer computes $\beta = b h b^{-1}, \gamma = b a_p^{-1} h a_p b^{-1}$ and sends (β, γ) to the user .

- `Unblinding`: User computes $\beta^{'} = \delta\beta\delta^{-1}$,$\gamma^{'} = \delta\gamma\delta^{-1}$ and display $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_0^{'})$ as a proxy blind signature on message *m*.

4.6 Balyan attack on the Unlinkability

In the proposed blind signature scheme, during the interactive protocol execution between the proxy signer and user,$(\alpha^{'}, \beta^{'}, \gamma^{'})$ is a signature on the message *m*. For the signer, in order to establish a link between revealed message $(\alpha^{'}, \beta^{'}, \gamma^{'}, t_0^{'}, m)$ and blind information $\alpha_i, \beta_i, \gamma_i$, the signer records owned all the generated information, such as $\alpha_i, \beta_i, \gamma_i$. After the signature $(\alpha_i^{'}, \beta_i^{'}, \gamma_i^{'}, t_0^{'}, m_i)$ is revealed , the signer executes the following steps:

1. Set a value β_i
2. Select a valid signature pair $(\alpha_i^{'}, \beta_i^{'}, \gamma_i^{'}, t_0^{'}, m_i)$.
3. Computes $h_i = H(H_1(t_0^{'} x_0^{'} || m)$.
4. Check the conjugacy relation $\beta_i \sim h_i$. if it is hold go to next step, otherwise go to step-1 and set a different value of β_i.
5. Set a value α_i
6. Check the conjugacy relation $\alpha_i \sim \alpha_i^{'}$, if it is hold go to next step, otherwise go to step-1 and set a different value of β_i.
7. Set a value γ_i.
8. Check the conjugacy relation $\gamma_i \sim h_i$. if it is hold go to next step, otherwise go to step-1 and set a different value of γ_i.

In the proxy blind signatue scheme [15], $\alpha_i = b_i x_p b_i^{-1}$, $\beta_i = b_i h_i b_i^{-1}$ and $\gamma_i = b_i a_p^{-1} h_i a_p b_i^{-1}$. On the otherside,$\alpha_i^{'} = \delta_i \alpha_i \delta_i^{-1}$,$\beta_i^{'} = \delta_i \beta_i \delta_i^{-1}$ and $\gamma_i^{'} = \delta_i \gamma_i \delta_i^{-1}$. It can be observed easily that every selected transcription $\alpha_i, \beta_i, \gamma_i$ will only be mapped uniquely on its corresponding transcription $(\alpha_i^{'}, \beta_i^{'}, \gamma_i^{'})$. In this way, the Verma's proposed proxy blind signature over braid group [15]is vulnerable to linkability attack and the signer is able to link a valid message signature $(\alpha_i^{'}, \beta_i^{'}, \gamma_i^{'}, m_i)$ with the blind information $\alpha_i, \beta_i, \gamma_i$.

5 Conclusions

This paper presents the security vulnerabilities of a proxy blind signature scheme over braid groups.It is clear that the proposed scheme does not satisfy two essential security peopoerties:unforgeability and unlinkability, which are two essential security requirement of a secure proxy blind signature scheme. Since, the delegation pair does not provide sufficient information about the original and proxy signer, therefore the proposed scheme has serious securities vulnerabilities. The author claimed that the proposed scheme is still secure even if the attacker intercept the delegation pair. This paper proved that this claim is not true and the proposed scheme is also vulnerable to the misuse of delegation pair, original signer changing attack and proxy signer changing attack.

References

[1] A. Boldyreva, A. Palacio and B. Warinschi, Secure proxy signature schemes for delegation of signing rights, available at *http://eprint.iacr.org/2003/096.*

[2] B. Lee, H. Kim, and K. Kim, Strong proxy signature and its applications, *in the Proceedings of SCIS2001*, pp. 603-608, 2001.

[3] Cao. T., Lin. D., and Xue. R., Improved privacy-protecting proxy signature scheme, Proc. of AWCC'04 - *Advanced Workshop on Content Computing*, Chi-Hung Chi, and Kwok-Yan Lam (Eds.), Volume 3309 of *LNCS*, pp.208-213, Spring-Verlag,2004.

[4] D. Chaum, Blind signature systems, Proceedings of Crypto 83, pp. 153- 158, Springer Verlag, 1984.

[5] D. Chaum, A. Fiat, M. Naor, Untraceable electronic cash, Proceedings of Crypto 88, LNCS - 403, pp. 319-327, Springer Verlag, 1988.

[6] D. Hofheinz and R. Steinwandt, A practical attack on some Braid group based cryptographic primitives, in *Public key Cryptography, PKC* 2003 proc., *LNCS* - 2567, pp. 187-198, Springer Verlag 2002.

[7] D. Pointcheval and J. Stern, Probably secure blind signature schemes, Proc. Asiacrypt-96, LNCS - 1163, pp. 252-265, Springer Verlag, 1996.

[8] Dai. J., Yang. X., and Dong. J., A privacy-protecting proxy signature scheme and its application, *Proc. of The 42nd annual Southeast regional conference, ACM* Southeast Regional Conference, pp.203-206, 2004.

[9] E. A. Elrifai and H. R. Morton, Algorithms for positive braids, *Quart. J. Math. Oxford* 45 (1994), 479-497.

[10] E. Lee, S. J. Lee and S. G. Hahn, Pseudorandomness from braid groups, *Advances in Cryptology*, Proceedings of Crypto 2001, *LNCS*- 2139, ed. J. Kilian, Springer-Verlag (2001), 486-502.

[11] Emil Artin, Theory of Braids, *Annals of Math*, 48, pp. 101-126, 1947.

[12] F. A. Garside, The braid group and other groups, *Quart. J. Math. Oxford* 20 (1969), no. 78, 235-254.

[13] F. Zhang and K. Kim. ID-Based Blind Signature and Ring Signature from Pairings. In Proceedings of ASIACRYPT 2002, LNCS-2501, pp. 533-547, Springer-Verlag, 2002.

[14] F. Zhang and K. Kim. Efficient ID-Based Blind Signature and Proxy Signature. In Proceedings of ACISP 2003, LNCS 2727, pp. 312-323, Springer-Verlag, 2003.

[15] G. K. Verma, A proxy blind signature schemes over Braid groups, International Journal of Network Security, Vol.9, No.3, pp. 214 - 217, Nov. 2009

[16] Guo. L., Wang. G., and Bao. F., On the security of a threshold proxy signature scheme using self-certified public keys, Proc. Of CISC'05 - *The SKLOIS conference on information security and cryptology*, Higher Education Press of China. Dec. 15-17, 2005.

[17] J. C. Cha, K. H. Ko, S. J. Lee, J. W. Van and J. S. Cheon, An efficient implementation of Braid groups, Proc. Of Asiacrypt-2001, *LNCS* -2248, pp. 144-156, Springer Verlag, 2001.

[18] J. S. Birman, Braids, links, and mapping class groups,*Annals of Math*, study 82, Princeton University Press (1974).

[19] J. S. Birman, K. H. Ko and S. J. Lee, A new approach to the word and conjugacy problem in the braid groups,*Advances in Mathematics* 139 (1998), 322-353.

[20] J. Zhang, T.Wei, J. Zhang and W. Zou. Linkability of a Blind Signature Scheme and Its Improved Scheme. In Proceedings of ICCSA 2006, LNCS 3983, pp. 262-270, Springer-Verlag, 2006.

[21] J. Zhang and W. Zou. Linkability of a Blind Signature Scheme. In Proceedings of ICICIC 2006, Vol. 1, pp. 468-471, IEEE, 2006.

[22] K. H. Ko, D. H. Choi, M. S. Cho and J. W. Han, New signature scheme using conjugacy problem, 2002, *available at http://eprint.iacr.org/2002/168.*

[23] K. H. Ko, S. J. Lee, J. H. Cheon, J. W. Han, J. S. Kang and C. S. Park, New public key cryptosystem using Braid groups, Proc. Crypto-2000, *LNCS*-1880, pp. 166- 183, Springer Verlag 2000.

[24] K. Manoj, Security analysis of a proxy signature scheme over Braid groups, *Cryptology eprint archieve report*, http://www.eprint.iacr.org/2009/158, 2009.

[25] K. Manoj, Linkability of Blind Signature Schemes over Braid Groups, *Cryptology eprint archieve report*, http://www.eprint.iacr.org/2009/192, 2009.

[26] K. Zhang, "Threshold proxy signature schemes," in the Proc of *Information Security Workshop*, Japan, pp. 191-197, 1997.

[27] L. Li., S.Tzeng, M. Hwang, : Generalization of proxy signature-based on discrete logarithms, *Computers & Security*, Vol. 22(3),pp.245-255, Elsevier Science, 2003.

[28] M. Abe and T. Okamoto. Provably Secure Partially Blind Signature. In Proceedings of CRYPTO 2000, LNCS 1880, pp. 271-286, Springer-Verlag, 2000.

[29] Mambo. M., Usuda. K., and Okamoto. E., Proxy signatures for delegating signing operation, Proc. of 3rd ACM *Conference on Computer and Communications Security*, pp.48-57, *ACM* press, 1996.

[30] S. H. Nagore and M. R. Sekhar, Nonrepudiable threshold proxy signatures with tracebility property, *Far East Journal of Applied Mathematics*, Vol. 6(3), pp. 233-240, 2002.

[31] S. J. Kim, S. J. Park, D. H. Won, Proxy Signatures, revisited, in the *Proceedings of ICICS*'97, *LNCS* - 1334, pp. 223-232, Springer-Verlag.

[32] S. Kim, S. Park and D. Won, Proxy signatures: Revisited, in Y. Han, T. Okamoto, S. Quing, editors, *Proceedings in International Conference on Information and Communications Security* (ICICS), of *LNCS*- 1334, pp 223-232, Springer Verlag, 1993.

[33] Sun. H., and Hsieh. B., On the security of some proxy signature schemes, *Cryptology ePrint Archive: Report 2003/068, available at http://eprint.iacr.org/2003/068.*

[34] Tan. Z., Liu. Z., and Wang. M., On the security of some nonrepudiable threshold proxy signature schemes, Proc. of ISPEC'05 - *First International Conference on Information Security Practice and Experience*, Robert H. Deng, Feng Bao, Hwee- Hwa Pang, and Jianying Zhou (Eds.), Volume 3439 of *LNCS*, pp.374-385. Springer-Verlag, 2005.

[35] Wang. G., Bao. F., Zhou. J., and Deng. R.H., Comments on a Threshold Proxy Signature Scheme Based on the RSA Cryptosystem, *Cryptology ePrint Archive: Report 2004/054, available at http://eprint.iacr.org/2004/054.*

[36] Wang. G., Bao. F., Zhou. J., and Deng. R.H., Security analysis of some proxy signatures, Proc. of ICISC'03 - *6th International Conference on Information Security and Cryptography*, Jong In Lim, and Dong Hoon Lee (Eds.), Volume 2971 of *LNCS*, pp.305-319, Springer-Verlag, 2003.

[37] Tan, Z., Liu, Z. and Tang, C. (2002), Digital proxy blind signature schemes based on DLP and ECDLP, in MM Research Preprints, No. 21, MMRC, AMSS, Academia, Sinica, Beijing, pp. 212217.

[38] Z. Huang, K. Chen, Y. Wang. Efficient Identity-Based Signatures and Blind Signatures. In Proceedings of CANS 2005, LNCS 3574, pp. 120-133, Springer-Verlag, 2005.

Manoj Kumar received the B.Sc. degree in mathematics from Meerut University Meerut, in 1993; the M. Sc. in Mathematics (Gold medalist) from C.C.S.University Meerut, in 1995; the M. Phil. (Gold medalist) in Cryptography, from Dr. B. R. A. University Agra, in 1996; the Ph.D. in Cryptography, in 2003. He also qualified the National Eligibility Test (NET), conducted by Council of Scientific and Industrial Research (CSIR), New Delhi- India, in 2000. He also taught applied Mathematics at D. A. V. College, Muzaffarnagar, India from Sep 1999 to March 2001; at S.D. College of Engineering & Technology, Muzaffarnagar- U.P. - INDIA from March 2001 to Nov 2001; at Hindustan College of Science & Technology, Farah, Mathura- U.P. - INDIA, from Nov 2001 to March 2005. In 2005, the Higher Education Commission of U.P. has selected him. Presently, he is working in Department of Mathematics, R. K. College Shamli- Muzaffarnagar- U.P. - INDIA-247776. He is a member of Indian Mathematical Society, Indian Society of Mathematics and Mathematical Science, Ramanujan Mathematical society, and Cryptography Research Society of India. He is working as a reviewer for some International peer review Journals: Journal of System and Software, Journal of Computer Security, International Journal of Network Security, The computer networks, computer and security, The Computer Journal. He is also working as a Technical Editor for some International peer review Journals- Asian Journal of Mathematics & Statistics, Asian Journal of Algebra, Trends in Applied Sciences Research, Journal of Applied Sciences. He has published his research works at National and International level. His current research interests include Cryptography and Applied Mathematics.