A Key Agreement Protocol Using Non-Abelian Group

Abhishek Dwivedi[1]
[1]Research Scholar Singhania University, Jhunjhunu, Rajsthan
& Department of M.C.A., Raj Kumar Goel Engineering College, Ghaziabad, U.P., India.
Email: dwivediabhi@gmail.com[1]
D.B.Ojha[2]
[2]Department of Mathematics, R. K. G. Institute of Technology, Ghaziabad, U.P., India.
Email: ojhdb@yahoo.co.in[2]

ABSTRACT

This paper presents a key agreement protocol based on a non abelian group. It is proved that the proposed protocol meets several security attributes under the assumption that the Root Problem (RP) in braid group is a hard problem.

Keywords - **Non abelian group, braid group, root problem, key agreement, and security.**

Date of Submission: October 28, 2010 Date of Acceptance: December 8, 2010

1. INTRODUCTION

The Key exchange problems are of central interest in security world. The basic aim is that two people who can only communicate via an insecure channel want to find a common secret key without any attack.

In this paper, we elaborated the process for well secured and assured for sanctity of correctness about the sender's and receiver's identity, as key agreement protocol under the root problem in non-abelian group (KAP-NAG).

In recent years have emerged as suitable settings for cryptographic protocols [5, 6, 7, and 8].The idea of using the braid group as a platform for cryptosystems was first introduced in 1999 by Anshel, Anshel and Goldfeld [7]. The useful feature of Braid groups is that they are more complicated than Abelian groups, but are not too complicated to work with. These two characteristics of braid group are useful to choose, whenever in search for good candidature, in this concern.

Root problem (RP) has been suggested by Sibert, Dehronoy, and Girault in 2003[5]. They also remarked that in open literature there is no cryptographic protocol based on RP. Here we use Root Problem to suggest a new key agreement scheme. Root Problem (RP) in braid groups is algorithmically difficult, and consequently provides one-way functions. We use it to propose a key agreement protocol over a braid group.

If sender and receiver both are in separate physically, they must trust a transmission medium to prevent the disclosure of the secret key being communicated. Anyone who intercepts the key in transit can later read, modify, and forge all messages encrypted using that key. The generation of such keys is called key agreement; and all cryptosystems must deal with key agreement issues. Because all keys in a symmetric cryptosystem must remain secret, secret-key cryptography often has difficulty providing secure key agreement, especially in open systems with a large number of users.

The concept of key agreement was introduced in 1976 by W. Diffie and M. Hellman [11]. In their seminal scheme each person gets a pair of keys, one called the public key and the other called the private key. Each person's public key is published while the private key is kept secret. The need for the sender and receiver to share secret information is thus eliminated; all communications involve only public keys, and no private key is ever transmitted or shared.

This paper is organized as follows: We present a brief introduction of braid groups in section 2. In section 3, we define key agreement protocol mention its desirable attributes. In section 4, we present our protocol. In section 5, the security consideration is mentioned. Finally ends with conclusion.

2. PRELIMINARIES

Braid Groups:

Emil Artin [4] in 1925 defined B_n, the braid group of index n, using following generators and relations: Consider the generators $\sigma_1, \sigma_2, \ldots . \sigma_n$, where σ_i represents the braid in which the $(i+1)^{st}$ string crosses over the ith string while all other strings remain uncrossed. The defining relations are

1. $\sigma_i\sigma_j = \sigma_j\sigma_i \ for \ |i-j| \geq 2,$

2. $\sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j \ for \ |i-j| = 1,$

An n-braid has the following geometric interpretation: It is a set of disjoint n-strands all of which are attached to two horizontal bars at the top and at the bottom such that each strands always heads downward as one walks along the strand from the top to the bottom. In this geometric interpretation, each generator ó$_i$ represents the process of swapping the i^{th} strand with the next one (with ith strand going under the $(i+1)^{th}$ one). Two braids are equivalent if one can be deformed to the other continuously in the set of braids. B_n is the set of all equivalence classes of geometric n-braids with a natural group structure. The multiplication ab of two braids a *and* b is the braid obtained by positioning a on the top of b. The identity e is the braid consisting of n straight vertical strands and the inverse of a is the reflection of a with respect to a horizontal line. So σ^{-1} can be obtained from σ by switching the over-strand and under-strand. $\Delta=$ $(\sigma_1, \sigma_2, \ldots\ldots \sigma_{n-1})(\sigma_1, \sigma_2, \ldots\ldots \sigma_{n-2}) \ldots\ldots (\sigma_1, \sigma_2)(\sigma_1)$ is called the fundamental braid.

If b is a non-trivial and $e \geq 2$ is an integer, then b^e is never identity. In other words, the braid groups are torsion free. The Root Problem in B_n is to find, given y and $e \geq 2$, an x such that $y = x^e$. It is proved in [10] that RP is decidable but is computationally infeasible if braids of a sufficient size are considered.

3. OUR PROPOSED SCHEME

In this section we describe our two pass key agreement protocol (KAP) between two entities $A\ and\ B$, and consider its security.

For our scheme, the initial setup known to both $A\ and\ B$ is a braid group B_n where RP is infeasible. As mentioned earlier, all the braids in B_n are assumed to be in the left canonical form. Thus for *a, b* in B_n, it is hard to guess a or b from ab. We assume that n is even, and denote by LB_n (resp.UB_n) the subgroup of B_n generated by $\sigma_1 \ldots. \sigma_{\frac{n}{2}-1}$, i.e., braids where the *n/2* lower strands only are braided (resp. in the subgroup generated by $\sigma_{\frac{n}{2}+1} \ldots. \sigma_{n-1}$). We know that every element in LB_n commutes with every element in UB_n . B_n is finite and non-commutative, so problem on braid group are non trivial.

We denote by

R	:	sufficiently complicated braid group
		V= $ID_A \| R$
		W=$ID_B \| R$

$S_{A_1}(V), S_{A_2}(V) \in LB_n$	:	$A's$ long term private key pair
$S_{A_1}^e(V)R\ S_{A_2}^e(V) = X_A$	:	$A's$ long term public key
$S_{B_1}(W), S_{B_2}(W) \in U_{B_n}$	:	$B's$ long term private key pair
$S_{B_1}^e(W)R\ S_{B_2}^e(W) = X_B$	:	$B's$ long term public key
$U_{A_1}^e(V), U_{A_2}^e(V) \in LB_n$	:	$U_{A_1}^e(V)R\ U_{A_2}^e(V) = Y_A$
$Z_{B_1}^e(W), Z_{B_2}^e(W) \in U_{B_n}$	:	$Z_{B_1}^e(W)R\ Z_{B_2}^e(W) = Y_B$
$S_{A_1}^e(V)X_b\ S_{A_2}^e(V)$	:	K_A
$S_{B_1}^e(W)X_a\ S_{B_2}^e(W)$	:	K_B
$K_B^e(Y_B)K_B^e = K_B^e Z_{B_1}^e(W)R\ Z_{B_2}^e(W)K_B^e$	:	Y_B
$U_{A_1}^e(V)\ \text{魔}_A^{-e}\ Z_{B_1}^e(W)R\ Z_{B_2}^e(W)K_A^{-e}U_{A_2}^e(V)$	:	$Key\ Y_A$
$Z_{B_1}^e(W)U_{A_1}^e(V)R\ U_{A_2}^e(V)Z_{B_2}^e(W) = K$	:	$Key\ Y_B$
h	:	strong one $-$ way hash function

4. KEY AGREEMENT

Here we describe the KAP-NAG following the above notations. The protocol works in the following steps.

1. A randomly chooses $U_{A_1}^e(V)\ and\ U_{A_2}^e(V)\ in\ LB_n$, computes $U_{A_1}^e(V)R\ U_{A_2}^e(V) = Y_A$ if $Y_A = I$ (Identity braid), A terminates the protocol and restarts with new $U_{A_1}^e(V)\ and\ U_{A_2}^e(V)$.
   A, then sends $h(Y_A)$ to B.

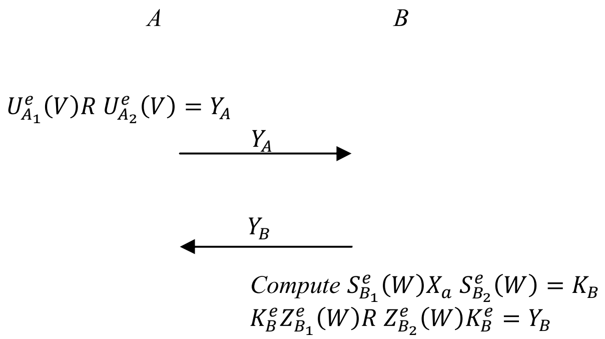
A
B
$U^e_{A_1}(V)R\ U^e_{A_2}(V) = Y_A$
Y_A
Y_B
Compute $S^e_{B_1}(W)X_a\ S^e_{B_2}(W) = K_B$
$K^e_B Z^e_{B_1}(W)R\ Z^e_{B_2}(W)K^e_B = Y_B$

Fig.1. Two-pass KAP-NAG Protocol

2. Upon receiving Y_A, B randomly chooses $Z^e_{B_1}(W)\ and\ Z^e_{B_2}(W)\ in\ UB_n$, computes $K_B = S^e_{B_1}(W)X_a\ S^e_{B_2}(W)$, and $Y_B = K^e_B Z^e_{B_1}(W)R\ Z^e_{B_2}(W)K^e_B$ or $Y_B = I$, B terminates the protocol and restarts with new $Z^e_{B_1}(W)\ and\ Z^e_{B_2}(W)$. B, then sends $h(Y_B)$ to A.
3. Upon receiving Y_B, A computes $K_B = S^e_{A_1}(V)X_b\ S^e_{A_2}(V) = K_A$, and the shared key $Key\ Y_A = U^e_{A_1}(V)K^{-e}_A Y_B K^{-e}_A U^e_{A_2}(V)$.
4. B also computes the shared key $Key\ Y_B = Z^e_{B_1}(W)Y_A Z^e_{B_2}(W) = K$.
5. After regular protocol running, $A\ and\ B$ share the secret $K = Key\ Y_A = Key\ Y_B$.

5. SECURITY CONSIDERATION

Here we show that our protocol meets the following desirable attributes under the assumption that the root problem is hard.

Known-Key Security: If $A\ and\ B$ execute the regular protocol run, they clearly share their unique session key 汜, because
$Key\ Y_A = U^e_{A_1}(V)K_{\_}^{-e}\ Y_B K^{-e}_A U^e_{A_2}(V)$
$= U^e_{A_1}(V)K^{-e}_A K^e_B Z^e_{B_1}(W)RZ^e_{B_2}(W)K^e_B K^{-e}_A U^e_{A_2}(V)$
$= U^e_{A_1}(V)Z^e_{B_1}(W)RZ^e_{B_2}(W)U^e_{A_2}(V)$
$= Z^e_{B_1}(W)U^e_{A_1}(V)R\ U^e_{A_2}(V)Z^e_{B_2}(W) =$
$Z^e_{B_1}(W)Y_A Z^e_{B_2}(W) = Key\ Y_B$.

(Perfect) Forward Secrecy: During the computation of the session key K for each entity, the random braids a_1, a_2, b_1, b_2 still act on it. An adversary who captured their private keys $U_{A_1}(V), U_{A_2}(V)$ or $Z_{B_1}(W), Z_{B_2}(W)$ should extract $K_A\ and\ K_B$from the information $Y_A\ and\ Y_B$ to know the previous or next session keys between them. However, this is the very root problem. Hence, under the assumption that the RP is computationally infeasible, NBKAPN meets the forward secrecy requirement.

Key - Compromise Impersonation: Suppose $\mathcal{U}'s$ long -term private key, $U_{A_1}(V), U_{A_2}(V)$, is disclosed. Now an adversary who knows this value can clearly impersonate A. Is it possible for the adversary impersonates $B\ to\ A$ without knowing the $B's$ long-term private key, $Z_{B_1}(W), Z_{B_2}(W)$? For the success of the impersonation, the adversary must know $A' -$ ephermeral key $U_{A_1}(V), U_{A_2}(V)$ at least. So, also in this case, the adversary should extract $U_{A_1}(V), U_{A_2}(V)$ from $A's$ ephemeral public value $Y_A = U^e_{A_1}(V)R\ U^e_{A_2}(V)$. This also contradicts that RP is hard.

Unknown Key-share: We examine the unknown key-share attack that allows an adversary E to make one party

believe K to be shared with E while it is in fact shared with a different party. A common scenario is that E has X_A certified without knowing the private key $U_{A_1}(V), U_{A_2}(V)$ of A, and uses it to talk with $B\ as\ E$ while she poses as $B\ to\ A$ simultaneously. Our protocol is secure against this attack because for E, we have

$h(Y_A)\ h(Y_B)$ in computing each K

Key Control: As the same argument in the above, the key-control is clearly impossible for the third party. The only possibility of *key-control* attack may be brought out by the participant of the protocol B. But for the entity B, to make the party, A generate the session key $K(Key\ Y_B)$. which is pre -selected value by B, for example B should solve the following $K = Key\ Y_A = Key\ Y_B$. But this again falls into the problem of RP.

6. CONCLUSION

Our key agreement protocols have quality for being a useful part of secure e-gaming and e-gambling protocols. In fact, our approach are a guarantee that no player misbehaviors or deviates from the protocols, because they agreed at one point. In this paper, we have presented a key agreement protocol that allows both players to agree at a bitstring in a non-repudiable way based on the root problems.

REFERENCES

[1] Atul Chaturvedi, Shyam Sundar ”*A Secure Key Agreement Protocol Using in Braid Groups”* in International Journal of Advanced Networking and Applications, Vol.01, Issue: 05, Pages: 327-330, 2010.

[2] B. Preneel, *“The state of cryptographic hash functions”* in Lectures on Data Security: Modern Cryptology in Theory and Practice, LNCS 1561, Berlin: Springer, pp. 158-192, 1999.

[3] D.B.Ojha, Abhishek Dwivedi, Ajay Sharma, & Ramveer Singh, *“A Non-Repudiable Biased Bitstring Key Agreement protocol (NBBKAP) Using Conjugacy Problem in Non-abelian Group”*, International Journal of Engineering Science and technology Vol. 2(9), 2010, 4162-4166.

[4] E. Artin, *“Theory of braids”*, Annals of Mathematics, Vol. 48, pp. 101-126, 1947.

[5] H.Sibert, P.Dehornoy, & M.Girault, *“Entity authentication schemes using braid word reduction”*, WCC 2003, to appear; http://eprint.iacr.org/2002/187.

[6] I.Anshel, M.Anshel, B.Fisher, and D.Goldfeld, *“New key agreement protocols in braid group cryptography”*, Proc.of CT – RSA 2001, LNCS, 2010, Springer-Verlag, 1-15.

[7] I. Anshel, M. Anshel and D. Goldfeld, *“An algebraic method of public-key cryptography”*, Math. Research Letters, 6, 1999, 287-291.

[8] K.H. Ko, S.J. Lee, J.H. Cheon, J.W. Han, J.S. Kang, and C Park, *“New public -key cryptosystem using braid groups, Advances in Cryptology”*, Proceeding of Crypto - 2000, Lecture Notes in Computer Science 1880, ed. M Bellore, springs Verlag, 2000, 166-183.

[9] M. Blum, *“Coin flipping by telephone: a protocol for solving impossible problems”*, Proc. IEEE Computer Conference, pp. 133-137, 1982.

[10] V.B.Styshnev, *“The extraction of a root in a braid group (English)”*, Math. USSR Izv. 13, 1979, 405-416.

[11] W.Diffie, & M.Hellman, *“New directions in cryptography"*, IEEE Trans. Inform. Theory, 22 (6), 1976, 644-654.

Authors Biography

Abhishek Dwivedi, Master of Computer Application from Uttar Pradesh Technical University, Lucknow (U.P.), India in 2007. Pursuing Ph.D from Singhania University, Jhunjhunu, Rajsthan, India. He has more than four year experience in teaching and research as Assistant Professor. He is working at Raj Kumar Goel Engineering College, Ghaziabad (U.P.), India. His main research interests are in Public Key Cryptography and its applications.

Dr. Deo Brat Ojha, Ph.D from Department of Applied Mathematics, Institute of Technology, Banaras Hindu University, Varanasi (U.P.), India in 2004. His research field is Optimization Techniques, Functional Analysis & Cryptography. He has more than Six year teaching & more than eight year research experience. . He is working as a Professor at Raj Kumar Goel Institute of Technology, Ghaziabad (U.P.), India. He is the author/co-author of more than 50 publications in International/National journals and conferences.