Pairwise Independent Key Generation Algorithm: A Survey

Nikita Gupta
M.E.(CSE) Scholar
Department of CSE
Truba Institute of Engineering & Information
Bhopal, India

Amit Saxena
Associate Prof. &Head
Department of CSE
Truba Institute of Engineering & Information Technology
Bhopal, India

Nakul Jain
B.E. Final Year
Department of Computer Engg.
Institute of Engineering and Technology DAVV
Indore; India

ABSTRACT
Key Generation is a technique to provide a secure and efficient generation of Key Pairs so that the keys can't be attacked by the external or unauthorized users. Since there are various techniques implemented for the generation of keys such as based on Graphical Methods [1]. Here in this paper a review of all the existing techniques implemented for the generation of Key Pairs is analyzed and discuss their various advantages and limitations, so that on the basis of various issues in the existing Key Pairs Generation Techniques a new and efficient technique is implemented in future.n this paper, we describe the formatting guidelines for IJCA Journal Submission.

Keywords
Network Security, Group Key Establishment, Contract Signing Protocol, Key Pair Generation Center, Key Pairwise Independent Networks.

1. INTRODUCTION
In any electronic transaction the two parties or more than two parties don't want to trust each other or each other transactions this is the reason why a type of signing protocol is needed in the situation which is known by a normal language a contract signing protocol. The contract signing is easy in paper based model due to existence of simultaneous. Two parties hard reproductions of the same agreement are approved or signed by both the parties at the identical time and at the identical place. After the contract ratification both of them are approved on that document. So, if one of them do not agree on that document or contract then the other one must provide the signed document in the court. Now a day's many business oriented application or business uses the electronic transactions, for electronic transactions we are using key transfer protocol. When we talk about paper based contract then the signing on that document is very necessary and both the person have to sign on that document at the same time and at the same place. If both the parties are unable to meet for signing on the contract then the scheme electronic signing contract is the next alternative. When both the parties having lack of trust between them then this scheme which is known as electronic contract signing is totally fail. Many time one party or one user may send their electronic signature to other party but in many ways the other person or party may not return the signature to that one party. For solving this problem we are using group key establishment scheme. With the help of this scheme we can establish a mutual meeting key which is known only by the authorized group member but not others for communication. For this we are using key transfer protocol. In this protocol we are using key cohort center (KGC) which is to generate session keys for communication.

1.1 Group Key Establishment
In order to benefit of protected group leaning requests, multiple users need to share a private key, which is obtained as the output of a Group Key Establishment (GKE) protocol.

The main area of GKE is to establish a common key between the authorized members of a group, without disclosing it to other parties. The authorized participants to the protocol are also addressed as qualified, legitimate or privileged. A protocol runs for multiple times, named sessions. Each meeting is exceptionally recognized by a session id, which can be computed during the performance of the protocol or given in advance by the environment. We call meeting key the shared secret derived after one execution of the protocol. It only persists for a small period of time, a natural approach in cryptography (the probability to reveal key increases with its period of usage). To become eligible to take part to protocol sessions, users must first register within the group. After registration, they acquire a long-lived or long-term secret, which they will later use to derive the session keys they are qualified for.Menezes and al. motivate the importance of GKE

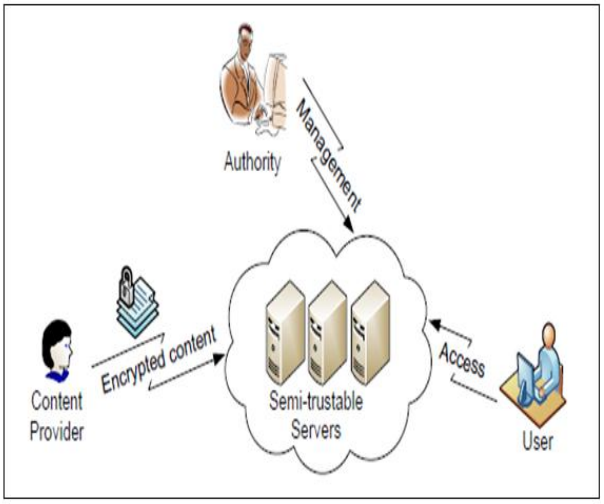

Fig 1: An Example Application Scenario of Secret Data Sharing

in addition to its main target (to establish the group key that is necessary to implement cryptographic properties, like confidentiality or group authentication), is:

- Limits the quantity of messages encrypted under the same key (by refreshing the assembly key for each session), which makes the system more powerful against cryptanalytic attacks;
- Restricts information disclosure in time if the key is compromised (for one session);
- Avoids the long-term stowing of a huge quantity of secret keys by creating keys at demand;
- Permits independence between communication sessions and applications.

1.1.1 Multiple Phases Of Gke Protocol (In General)

1.1.1.1 Initialization

It defines the environment of the protocol: the parameters, the space of all possible keys and any other prerequisites.

1.1.1.2 Users Registration

It assigns group association to users. Depending on the scenario, after registration, a user can do certain things for example share a secret key (or password) with a trusted group authority or may generate a certified long-lived public-private important couple for later signing purposes.

1.1.1.3 Execution

It describes the cryptographic algorithm, including the performed computations and the exchanged messages. It usually consists of multiple rounds of communication between principals.

1.1.1.4 Key Computation

It explicit the key computation formulas or algorithms performed by a party to derive the key from the information he gained after the Execution Phase. It is sometimes integrated within a round of the execution phase.

1.1.1.5 Key Confirmation:

It confirms that all the intended members actually own the key and no other except them does. Although it is an optional phase, it is usually performed for security reasons.

1.1.2 Classification

GKE rule partition into two module: assembly Key Transport (GKT) and Group Key Agreement (GKA). The main difference connecting the two lessons derives unswervingly from their definitions.

1.1.2.1 Group Key Transport (GKT):

GKT requires the existence of a privileged party to select and distribute the key, while GKA does not, the key being computed as the result of the collaboration of legitimate participants via exchanged messages. GKT permits the entity that generates the key to be an outsider as well (i.e. not a group member). This entity has various names in the literature, such as: Trusted Third Party (TTP), Key Generation Center (KGC), Key Distribution Center (KDC) or Group Controller [2], [3]. The naming differs according to the precise function it fulfills. For example, it may exist an entity that generates the key (KGC) and an entity (distinct or not) that allocates it to the authorized members (KDC). For the rest of this work we will mainly refer to the KGC as a single party that performs both key generation and distribution.

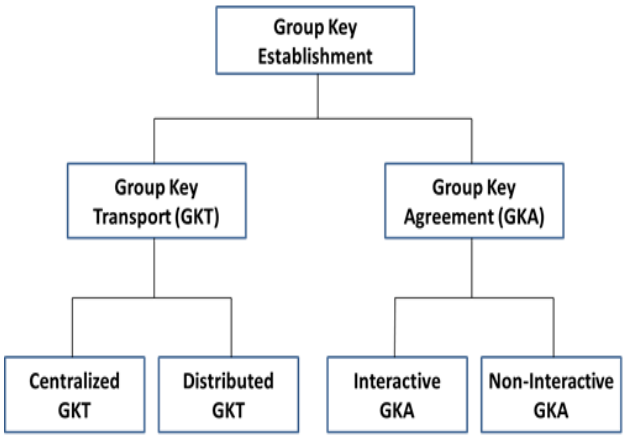

Fig 2: Classification of Group Key Establishment

The KGC must be trusted by all participants as honest in the sense that it selects a fresh key (a consistently accidental worth that has certainly not been used before) and does not reveal it to unqualified parties. GKT assumes (in general) the continuation of protected communiqué channels between the KGC and each user in the Users Registration Phase: the long-lived key of a participant frequently consists in a pre-shared secret (symmetric key or password) with the KGC. GKT protocols are primary used in application with centralized control. Based on the particularity of the entity that produces and allocates the key, GKT can further divide into various categories [4]:

Centralized GKT:

It contains a solitary individual that foodstuffs and apportions the crucial. Some of the drawbacks of this category include [4]

- The KGC must be always online
- The KGC must maintain a protected communique channel with each group member;
- The KGC may easily be the target of a DoS attack; (4) the computational power of the KGC limits the quantity of operators he can handle.

Distributed GKT:

It contains a single object that generates the key, while the distribution is performed by one of the qualified members of the group, which is dynamically selected for each execution of the protocol. Although this category is more suitable (especially for unreliable networks), it preserves the first two drawbacks of the Centralized GKT and only diminishes the last two. As a disadvantage, we mention that the construction and maintenance of such structures becomes more complicated (especially in case of dynamic groups). On the contrary, GKA protocols are not limited by the previously specified disadvantages of GKT: they do not provide a single point of trust, are more robust and the computational cost is in general balanced between all the participants to the protocol.

1.1.2.2 Group Key Agreement (GKA):

GKA doesn't require the existence of a privileged party to select and distribute the key, in GKA the key is derived only by the cooperation of internal group members The trust for the entity (KGC, Trusted Third Party) is not required for GKA protocols, which do not demand the existence of a privileged party to decide on the solution, but subtract it by equal donation of the principals. However, regardless of the GKE

type, a conviction relative is compulsory: the competent participants to a meeting trust each other, that none of them discloses the collective key. or else, the discretion of the set of rules is despoiled by evasion.

We remark that during the execution of a GKA protocol, participants do not trust each other and suspect their partners may intend to get control over the group key value. Due to less trust assumptions, GKA usually satisfies stronger security. Like GKT assumes, the long-lived key of the participant frequently consists in a pre-shared secret (symmetric key or password) with the KGC, GKA do not impose such an assumption: the long-lived keys of group members are usually public-private pair's uses for signing (or sometimes, for asymmetric encryption). Regarding the contribution type of the participants to the GKA (a nonce or the long-lived key), GKA split into [2]:

Interactive GKA:
Group members contribute to the key generation with fresh values for each session (nonce). They require exchanged messages between the participants and therefore impose that all parties are online for the execution of the protocol.

Non-Interactive GKA:
Group members contribute to the key generation with their own public long-live keys. Examples include the original Diffie-Hellman protocol [5] and Joux tripartite protocol [6]. Unlike the Interactive GKA, their main advantage is that a user can determine the common key even if the others are offline.

1.1.3 GKE Based on Clandestine Distribution
Secret sharing is used in GKE protocols to avoid such disadvantages:

- Allowing efficient constructions.
- Users may communicate through broadcast channels only.
- The computation of the key may consist in simple linear equations.
- The number of rounds remains constant regardless the group size.

In addition, they introduce several benefits:

- A convenient way to differentiate between principal's power within the group.
- Delegation of duties by passing shares to other participants. - Collection confirmation as an alternative of thing verification.
- Cheating detection and simple organization of cluster sizing using the customary doorstep [7].

1.2 Secure Key Establishment
Secure communiqué over computer networks is usually achieved by means of encrypting the exchanged messages. The messages could be encrypted by means of long-term public keys (or long-term shared keys). However, the last case would require that they share the same clandestine important which can be achieved by means of some secure key establishment protocol.

By means of such protocols, two or more individuals can establish shared secret cryptographic keys over unconfidential networks. The protocols can be based on secret key cryptography or public key cryptography. Due to sharing of long-term underground answers among a quantity of operators is an unreasonable assumption, most key formation etiquettes that is based on shared key (a.k.a. symmetric key) cryptography require an online TTP. Hence, each user would share a secret key with the TTP, and all key establishment messages would go through the TTP. Kerberos [9] and the symmetric important etiquette of Needham-Schroeder [8] are two well-known examples of key establishment etiquettes based on shared secret keys.

As it would be a problem to distribute and establish new shared keys to new users over an insecure network if a key is not already shared between the new user and the TTP. The advantage of public key ciphers is simplification of key management and eliminating the need for an online TTP. This increases considerably the usability for protocols based on public key ciphers, which have therefore become far more important than symmetric key protocols. Most public key protocols are based on a few well-known problems in number theory like the Discrete Logarithm Problem, the closely connected Diffie-Hellman Problematic, and the Factorization Problem (i.e., the difficulty of factorizing integers composed of two very large primes). For example, the RSA public important cryptosystem [13] is based on the Factorization Problem, and the ELGamal community key cryptosystem [16] is based on the two closely lined Diffie-Hellman Problematic and the Separate Logarithm Tricky. All security protocols in this thesis are public key-based. Key founding etiquettes can basically be divided into key transmission procedure and key arrangement protocols. Key transfer is where one entity generates the secret key and distributes it confidentially to one or more users. Key agreement is where two or more participants that "agree" on a secret key by equally contributing to the value of the established key. Rendering to the quantity of participants, such protocols are categorized as two-party and multi-party protocols.

1.3 Informal Security Requirements
A GKE protocol ought to gratify a set of properties, which we casually remember next. Key confidentiality (also called key privacy, key secrecy or non-disclosure) [11], [12] guarantees that it is (computationally) infeasible for an adversary to compute the cluster solution The stronger concept of acknowledged key sanctuary assures that key confidentiality is maintained even if the aggressor someway manages to attain group keys of preceding sessions.

Backward secrecy [10] conserves the privacy of expectations keys despite the adversary's proceedings in the precedent sessions. Correspondingly, forward secrecy [10] imposes that the challenger proceedings in outlook runs of the procedure do not negotiation the privacy of previous session keys (i.e. a key remnants protected in the prospect). Input assortment must satisfy specific properties. Key freshness requires that the collection important has never been used before. The related concept of key independence imposes that no correlation exists between keys from dissimilar sessions; this earnings that (collaboration flanked by) authorized participants to distinct sessions of the technique cannot unveil meeting keys they are unlawful for. In addition, key randomness warrants key in-distinguish ability from an accidental quantity and hence key unpredictability. Two other important security requirements regarding the key value exist: key integrity which attests that no adversary can modify the group key and key consistency, which prevents dissimilar company to recognize dissimilar keys.

Collection associate authentication represents a mandatory condition for group cryptographic protocols. Entity authentication confirms the identity of a participant to the protocol to the others. Similarly, unknown key share resilience restricts a user to believe that the key is shared with one party when in information it is communal with a different. Key cooperation impersonation (KCI) resilience [15] prevents an attacker who owns the long-lived input of a contributor to imitate other party to him. The stronger property named ephemeral key leakage (EKL) resilience (EKL) [16] avoids an adversary to recover the group key even if he discloses the long-lived keys and passing keys of parties implicated excluding both these standards for participants in the test session1. (Implicit) Key authentication restrictions the promising owner of the assemblage solution to the justifiable participants; this means that no other party except the competent users is accomplished to calculate the key, but it does not necessary mean that all legitimate principals actually own it. Another property, called key confirmation certifies that all authorized members actually have the key; however, it does not claim that no other party owns the same key. Explicit key authentication (or Mutual Authentication (MA)) [14], [10] combines these notions and ensures that all qualified participants to the protocol have actually computed the group key and no one else except them have. .

2. RELATED WORK

Lifeng Lai et. al's proposed a new efficient protocol for Pairwise Key generation over an Independent Networks using Graphical Model [1].

K. Kalaivaniet. al's uses Pairwise Independent Networks for Key Generation [17]. An Efficient Two Secrete Key generation for low complexity using indigenous key cohort and global propagation is proposed which provides better performance. Complex Algorithm for key generation and hence take more computational time. Sirin Nitinawaratet. al's implemented a Secrete Key Generation for the Generation of Pairwise Independent Networks [18].

Peng Xu et. al's proposed a new and efficient secluded Key volume using Cooperative Independent Key Pairwise Networks [19]. The detached is to produce a private key bordered by Alice and Bob underneath the help of the M relays; such a secluded key requirements to be endangered not only from Eve but also from individual relays simultaneously. High storage capacity for secrete keys is required.

Peng Xu et. al's also proposed M-Relay capacity based Pairwise key Generation for Private Keys [20].

Alfin Abraham, Vinodh Ewards, Harlay Maria Mathew [21] has proposed a review of the whole agreement validation etiquette that are previously been implement and planned. Here in this paper the main survey is on fairness an optimistic occupied throughout the indenture signing of the two parties so that if any party disastrous to get the autograph then the other festivity also doesn't get the autograph of the other party. A valid digital signature gives a recipient reason to consider that the communication was fashioned by a recognized dispatcher, and that it was not altered in shipment. Digital signatures are frequently used to execute electronic signatures, a larger term that mentions to any microelectronic statistics that transports the meaning of an autograph, but not all electronic autograph use alphanumeric autographs[1].

Alfin Abraham [11] has proposed a new-fangled and well-organized practice for the free fair swap over of agreement signing etiquette flanked by two parties. Here in this document an mistreatment free agreement procedure is given using the concept of generation of digital signatures by RSA technique. A succinct learning of the blond positive protocols

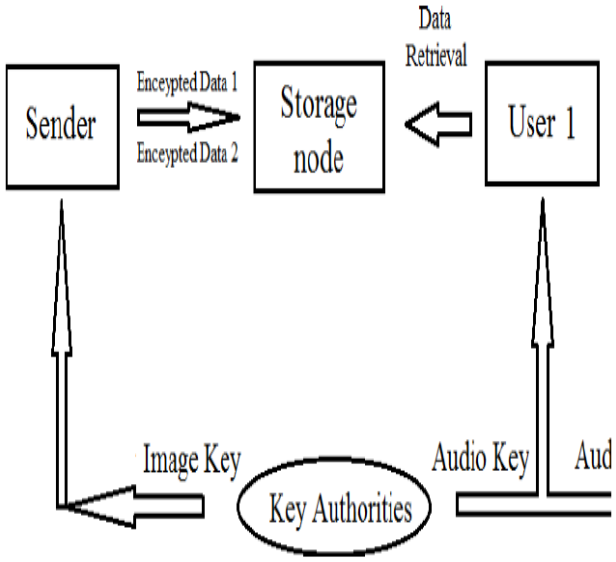

Fig.3. Flow of the methodology implemented in [17].

exchanging digital signatures is carried down and the analysis of the indispensable attribute, sanctuary and competence of the protocols is performed [22].

Guilin Wang [23] has proposed an abuse free fairness grounded contract validation procedure where the signature generation and corroboration can be finished using RSA method. Here in this three characteristic for the agreement signing over the internet is proposed i.e. it is abuse free, optimistic and involve TTP for the party to be cheated. The proposed algorithm is sheltered and is proficient in deference of time difficulty. As the electronic commerce is becoming more and more significant and accepted in the world, it is attractive to need a mechanism that allows two parties to sign a digital agreement via the Internet. However, the difficulty of agreement signing becomes difficult in this setting, since there is no simultaneity any additional in the situation of processor nets. In other words, the simultaneity has to be mimicked in order to intend a digital agreement signing procedure. This obligation is fundamentally captured by the concept of fairness: At the end of the procedure, moreover both parties have applicable signatures for a agreement or neither does, even if one of them tries to cheat or the communication channel is out of order. This is based on the standard RSA signature scheme; they proposed a innovative digital indenture signing procedure that allows two hypothetically doubted parties to exchange their digital signatures on a convention in an resourceful and protected way. Like the existing RSA-based solutions, the new protocol is fair-haired and cheerful, i.e., two parties get or do not get the other's digital signature simultaneously, and the important third party is only desirable in nonstandard suitcases that occur occasionally. However, diverse from all preceding RSA-based contracts validation etiquette, the planned etiquette is additinal abuse-free. Technical details are provided to show that our protocol meets a number of desirable properties, not only those just mentioned [23].

Table 1: Comparsion among various key generation techniques

S. N.	Paper	Author/Publication	Technique Used	Issues
1.	Key Generation Algorithms for Pairwise Independent Networks Based on Graphical Models.	Lifeng Lai, Siu-Wai Ho, IEEE Transaction on Information Thoery, 2015.	The two main components i.e. resident key cohort and comprehensive key dissemination is implemented. Local Key Generation is used for Point-to-point foundation coding with side material from which diagram can be raised and comprehensive key propagation is used to deliver various secrete keys.	Complex Algorithm for key generation and hence take more computational time.
2.	Pairwise Independent Network using Key Generation Algorithm.	K. Kalaivani, K. Renugadevi, Nithya, IOSR Journal of Computer Engineering, 2016.	An Efficient Two Secrete Key generation for low complexity using local key generation and global propagation is proposed which provides better performance.	Complex Algorithm for key generation and hence take more computational time.
3.	Secrete Key Generation for a Pairwise Independent Network Model.	SirinNitinawarat, Chunxuan Ye, Alexander Barg, IEEE Transactions on Information Theory, 2010.	The unbiased is to engender a undisclosed key collective by a given subcategory of terminuses at the principal rate probable, with the collaboration of any outstanding terminuses. A (single-letter) formula for clandestine important volume brings out a ordinary assembly amongst the problematic of underground key group and a combinatorial problematic of greatest stuffing of Steiner trees in an related multigraph.	High Storage Cost and Inefficient key generation.
4.	The private Key Capacity of a Cooperative Pairwise-Independent Network.	Peng Xu, Zhinguo Ding, Xuchu Dai, 2015.	In this broadside associated foundations pragmatic by every pair of terminuses are self-determining of those foundations pragmatic by any other pair of mortal. All the termini can transfer with each other over a communal conduit which is also experimental by Eve quietly. The detached is to produce a isolated key amongst Alice and Bob under the help of the M communicates; such a isolated key desires to be dwindling not only from Eve but also from separate relays instantaneously.	High storage capacity for secrete keys is required.
5.	On the Private Key Capacity of the M-Relay Pairwise Independent Network.	Peng Xu, Zhinguo Ding, Xuchu Dai, IEEE International Symposium on Information Theory, 2015.	Connected foundations experimental by every pair of depots are autonomous of those foundations detected by any supplementary pair of lethal. All depots can transfer with each other over a communal waterway which is also practical by Eve noiselessly. The impartial is to produce a private key among Alice and Bob under the help of the M communicates.	High storage capacity for secrete keys is required.

Giuseppe Ateniese proposed a new technique of verifiable encryption based agreement signing procedure for the fair exchange of data over the internet amongst two gatherings. Here in this paper both the parties produce a autograph of the bond and distributing bit-by-bit their signatures to each supplementary. These etiquettes may be used as construction

blocks for designing well-organized fair communication of digital signatures. They have slightly modified the model of fair exchange by introducing an initialization phase for some of the digital signature schemes. However, this phase is done only once and the resulting protocols are much more efficient than those of prior art.

Jung Min Park et al [24] has also proposed a protocol of contract signing using RSA signatures but for the E-commerce via Distributed Systems. There are various applications running over internet that needs a fair exchange of information such as E-commerce. Here in these techniques uses multi-signatures using RSA. This scheme uses multi-signatures that are like-minded with the fundamental typical signature scheme that is used to integrate the fair-exchange feature with existing e-commerce systems. Zero-knowledge proofs are not used in the exchange protocol, of this approach which significantly increases efficiency.

Bao et. al.'s Fair Contract Signing Protocol [25] has proposed distributed exchange of parties for the agreement signing and their pledges to a agreement in a fair way such that either each of them can obtain the other's commitment, or neither of them does.

A real-world and wel-organized method for fair agreement validation is using an indistinguishable important third party. This agreement signing etiquette [25] conserves equality while lingering optimistic in the sense that the trusted party need not be complicated in the etiquette unless a certain dispute occurs. Compared with the protocols already implemented, this protocol is very efficient since only several basic cryptographic operations are required. This protocol is more efficient as compared to the other fairness protocol such as Micali's [26] protocol.

3. CONCLUSION

Here in this paper a survey of all the existing technique implemented for Key Pairwise Generation over Independent Networks is analysed and discuss. The Survey of all the existing technique provides advantages and limitations of the existing techniques so that the efficient technique is implemented in future

4. REFERENCES

[1] Lifeng Lai, Siu-Wai Ho," Key Generation Algorithms for Pairwise Independent Networks Based on Graphical Models", IEEE Transactions on Information Theory, 2015.

[2] Yair Amir, Yongdae Kim, Cristina Nita-Rotaru, and Gene Tsudik. On the performance of group key agreement protocols. ACM Trans. Inf. Syst. Secur., 7(3):457{488, August 2004. pages 47.

[3] Alfred J. Menezes, Scott A. Vanstone, and Paul C. Van Oorschot. Handbook of Applied Cryptography. CRC Press, Inc., 1996. pages 21, 22, 45, 46, 47, 48, 51, 52, 99, 105.

[4] Jose A. Onieva1, Jianying Zhou, and Javier Lopez "Analysis of an Asynchronous Multi Party Contract Signing Protocol", Progress in Cryptology - INDOCRYPT 2005, Lecture Notes in Computer Science, Volume 3797, pp 311-321, and 2005.

[5] Giuseppe Ateniese, "Efficient Verifiable Encryption (and Fair Exchange) of Digital Signatures", Proceedings of the 6th ACM conference on Computer and communications security, pp. 138 – 146, ACM 1999.

[6] F. Bao, R. H. Deng, and W. Mao, "Efficient and practical fair exchange protocols with off-line TTP," in Proc. IEEE Symp. Security and Privacy, pp. 77–85, 1998.

[7] J. M. Park, E. Chong, H. J. Siegel, and I. Ray "Constructing fair exchange protocols for e-commerce via distributed computation of RSA signatures," in Proceedings PODC'03, pp. 172–181, ACM Press, 2003.

[8] G. Wang, "Generic non-repudiation protocols supporting transparent off-line TTP," Journal of Computer Security, vol. 14, no. 5, pp. 441–467, Nov. 2006

[9] Vinod Moreshwar Vaze," Digital Signature on-line, One Time Private Key [OTPK]", International Journal of Scientific & Engineering Research, ISSN:2229-5518, Volume 3, Issue 3, March -2012.

[10] Guilin Wang. "An Abuse-Free Fair Contract-Signing Protocol Based on the RSA Signature", IEEE Transactions On Information Forensics And Security, Vol. 5, No. 1, March 2010.

[11] Alfin Abraham, "An Abuse-Free Optimistic Contract Signing Protocol with Multiple TTPs", IJCA Special Issue on "Computational Science – New Dimensions & Perspectives" NCCSE, 2011.

[12] Giuseppe Ateniese, "Efficient Verifiable Encryption (and Fair Exchange) of Digital Signatures", Proceedings of the 6th ACM conference on Computer and communications security, pp. 138 – 146, ACM 1999.

[13] Emmanuel Bresson, Olivier Chevassut, and David Pointcheval. Provably authenticated group Diffie-Hellman key exchange - The dynamic case. In Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology, ASIACRYPT '01, pages 290{309, London, UK, 2001. Springer-Verlag. pages 52, 79, 87

[14] N. Asokan, V. Shoup, and M. Waidner, "Optimistic fair exchange of digital signatures," IEEE J. Sel. Areas Commun., vol. 18, no. 4, pp. 591–606, Apr.2000.

[15] F. Bao, G. Wang, J. Zhou, and H. Zhu, "Analysis and improvement of Micali's fair contract signing protocol," in Proc. ACISP'04, vol. 3108, LNCS, pp. 176–187, Springer-Verlag, 2004.

[16] J. Garay, M. Jakobsson, and P. MacKenzie, "Abuse-free optimistic contract signing," in Proc. CRYPTO'99, vol. 1666, LNCS, pp. 449 – 466, Springer-Verlag, 1999.

[17] K. Kalaivani, K. Renugadevi, Nithya," Pairwise Independent Network using Key Generation Algorithm", IOSR Journal of Computer Engineering, 2016.

[18] Sirin Nitinawarat, Chunxuan Ye, Alexander Barg," Secrete Key Generation for a Pairwise Independent Network Model", IEEE Transactions on Information Theory, 2010.

[19] Peng Xu, Zhinguo Ding, Xuchu Dai," The private Key Capacity of a Cooperative Pairwise-Independent Network", 2015.

[20] Peng Xu, Zhinguo Ding, Xuchu Dai," On the Private Key Capacity of the M-Relay Pairwise Independent Network", IEEE International Symposium on Information Theory, 2015.

[21] Alfin Abraham, Vinodh Ewards, Harlay Maria Mathew "A Survey on Optimistic Fair Digital Signature Exchange Protocols", International Journal on Computer Science and Engineering (IJCSE), ISSN: 0975-3397, Vol. 3, No. 2, pp. 821 – 825, Feb 2011.

[22] Mihhail Aizatulin, Henning Schnoor, and Thomas Wilke "Computationally Sound Analysis of a Probabilistic Contract Signing Protocol", Computer Security – ESORICS 2009, Lecture Notes in Computer Science, Volume 5789, pp. 571 - 586, and 2009.

[23] Ying Zhang, Chenyi Zhang, Jun Pang and Sjouke Mauw "Game-Based Verification of Multi-Party Contract Signing Protocols", Formal Aspects in Security and Trust, Lecture Notes in Computer Science, Volume 5983, pp 186-200, 2010.

[24] W. Diffie and M.E. Hellman, "New Directions in Cryptography," IEEE Trans. Information Theory, vol. IT-22, no. 6, pp. 644-654, Nov. 1976.

[25] Vinod Moreshwar Vaze," Digital Signature on-line, One Time Private Key [OTPK]", International Journal of Scientific& Engineering Research Volume 3, Issue 3, March -2012 1 ISSN22295518".

[26] H. Pagnia, H. Vogt and F. C. Gartner," Fair Exchange" The Computer Journal, 2003.