JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Password Authentication Key Exchange Protocol Review Paper

Ms. Sonam Tamrkar, Ms.Charu Jain Mr. Rajneesh Pachouri, Mr. Anurag Jain,
M Tech Scholar, Assistant Professor, Assistant Professor, Assistant Professor, Assistant Professor
Department of Computer Science & Engineering
Adina Institute of Science & Technology, Sagar, India

Abstract: *We employ a straightforward configuration of a fish tank as the changeable environment, recording its photos over time, in order to limit the computational complexity. To create the initial seed, the image data is then submitted to a reduction method and hash function. In order to get true random numbers, we suggest a method for economically removing the true seed from the image data and applying it to a pseudo-random generator, in this case, a Linear Congruential Generator (LCG). The need for information security is increasing across a range of industries, and security concerns are receiving more and more attention as a result of the Internet's explosive growth in recent years. Any future value cannot be accurately anticipated based on the current set of values and random numbers are those that are a part of a sequence in which values are uniformly distributed over a defined set. A safe session key can be provided to a pair of users interacting over an unstable channel using password-based encrypted key exchange protocols, even if the top secret key or password they share is chosen at random from a tiny set of values. We offer two straightforward Bellovin and Merritt-based encrypted key exchange systems.*

IndexTerms **- True random number generator, Image data**

I. INTRODUCTION

The encryption algorithm and key generation are crucial mechanism of an encryption scheme in the realm of information security; they must be unexpected [1-3]. Random numbers are an essential component of most cryptographic algorithms, and random number generators (RNG) have numerous uses in the field of information security, including creating the parameters for public key cryptosystems (such ECC and RSA) and encrypting images [5]. Figure 1 illustrates the division of random numbers into two groups, true random numbers (TRNs) and pseudo-random numbers (PRNs), based on the various random sequences that can be formed. In the context of simulation and testing, PRNs [6] refer to the extension of one seed into another lengthy output sequence by a predetermined method. They are typically repeatable. TRNs, in contrast to PRNs, can only be produced by random physical processes; they cannot be produced by pure mathematical random methods. TRNs have superior statistical properties compared to PRNs, as well as superior unpredictability. could be applied to systems with demanding security standards.

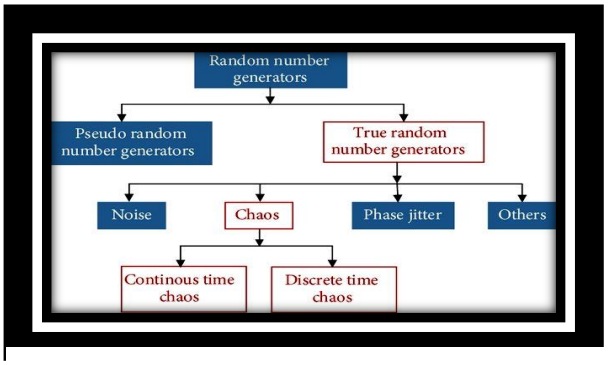

Figure 1 Architecture of random numbers generator.

Any future value cannot be accurately anticipated based on the current set of values and random numbers are those that are a part of a sequence in which values are homogeneously distributed over a defined set. The sequence is generated using a random number generator, as the term is commonly known. The terms "pseudo-random number generator" and "real random number generator" refer to two different categories of random number generators. A pseudo-random number generator employs a deterministic technique to generate random numbers from a seed value. In reality, the majority of sources for random numbers use

a pseudo-random generator. The actual entropy of the output can never be greater than the seed entropy because it is just a function of the seed data [2]. The worth of random numbers is crucial in some real-world safe systems. It is not advised to utilise PRNGs for these applications because they employ a deterministic method. Given that the output of these PRNGs is constant for a given seed, it follows that they cannot be safe if the seed is predictable in any way.

Figure 1 illustrates the three most important categories of entropy sources found in true random number generators (TRNGs) [7]: thermal noise on resistors and capacitors, phase jitter of oscillating signals, chaos, and others. The opposition sound for the thermal noise-based TRNGs is amplified to a respectable level. The digital random signal is produced by comparing the amplified noise voltage with the reference level using an ideal amplifier and a comparator. Because of the impact of some subpar Key exchange protocols are cryptographic primitives that are used to communicate a secure session key between two users via a dubious public channel.

Each has unique benefits and drawbacks. The SIGMA protocol [8], which serves as the foundation for the Internet Key Exchange (IKE) protocol's signature-based modes, is an illustration of a well-liked one. The 2-party symmetric setup, in which every pair of users shares a secret key, is the one that interests us in this study. We focus in particular on the case where the secret key is a password. key exchange using a password. Password-based key exchange protocols imply a more realistic scenario in which secret keys are selected from a restricted range of potential values rather than being randomly spread over a broad area (a four-digit pin, for example). Human-memorable passwords are also easier to use than, say, new cryptographic devices that can store high-entropy secret keys, so they seem more practical. However, the great majority of protocols used in reality do not take such a situation into consideration and are frequently vulnerable to "dictionary attacks." Dictionary attacks are attempts by an adversary to use a brute-force method to compromise a scheme's security by trying every combination of secret keys in a predetermined narrow set of values (i.e., the dictionary). These attacks can be quite harmful when the secret key is a password even though they are not very effective when applied to high-entropy keys since the attacker has a non-zero chance of succeeding. Several protocols that have been created to solve this issue are secure even when the secret key is a password. These methods are designed to limit the adversary's ability to succeed to online guessing attacks exclusively. The adversary must be present during these attacks in order to interact with the system and determine whether the system's guess is accurate. These systems' security often relies on a rule that invalidates or prohibits the use of a password after a predetermined number of failed attempts.

Encrypted key exchange. The encrypted key exchange (EKE) protocol developed by Bellovin and Merritt [8] is considered a landmark contribution in the field of password-based key exchange. Their protocol encrypts the Differ-Hellman key exchange protocol, which is executed by two users. Each flow is encrypted using the password that the two users share as the symmetric key. Numerous alternative protocols were suggested in the literatures that were based on theirs because of how straightforward it was, and each one had its own instantiation of the encryption function. Their EKE protocol is similar to our system. reducing the reliance on arbitrary oracles. One of our key objectives is to offer efficient and straightforward solutions that rely as little as possible on random oracles. The KOY protocol achieves this by fully eliminating the need for random oracles. These protocols, however, are frequently less effective than those built on the EKE protocol developed by Bellovin and Merritt [9].

II. RELATED WORK

Any business transaction involves a degree of mutual distrust between the participants. A contract signature is required in these circumstances. Using this protocol, two parties can quickly sign a digital contract over the Internet [9]. The exchange of digital signatures on a contract between two untrusted parties is made possible by a fair contract signing protocol. The initiator's private key is split into two pieces in this situation to ensure fairness, with one part going to the TTPhold and the other remaining secret. The initiator is the owner of the two parts of the private key. This digital contract signing process is based on the RSA signature and is secure because it only involves the third party in cases of fraud or a broken communication channel. In addition, if the protocol is not followed, none of the two parties can show the validity of intermediate results to other parties. As a result, the protocol is abuse-free. The trapdoor commitment method, a cryptographic primitive, is used to ensure abuse-freeness. A key security criterion for contract-signing procedures is the lack of abuse, especially in situations where committing just partially to a contract can benefit an untrustworthy party or an outsider.

Li et al. [9] and Yoon et al. [10] In view of elliptic curve encryption progress, we proposed two password-authenticated key exchange protocols without the server's public key. They claimed to provide security against a variety of potential attacks, easy user password updates, and explicit key authentication in the event of a session key agreement. The protocol created by Li et al. [10] is sadly vulnerable to offline dictionary attacks and man-in-the-middle attacks. Yoon et alprotocol, .'s on the other hand, lacks backward secrecy and is susceptible to off-line dictionary assaults. The flaws are carefully examined in this work by Junhan YANG and Tianjie CAO, who also offer a password-authenticated key exchange protocol that is secure against a variety of well-known attacks.

Key Differences	Symmetric Cryptography	Asymmetric Cryptography
cypher text size	Compared to the original plain text file, the encrypted text is smaller.	Compared to the original plain text file, the encrypted text is larger.
Size of Data	used to send bulky amounts of data	used to send little amounts of data
Utilization of Resources	Symmetric key encryption uses few resources to operate	High resource consumption is necessary for asymmetric encryption
Key Lengths	128 or 256-bit key lengths are available.	RSA key size of 2048 bits or more.
Security	Less safe because just one key is used for encryption	Much safer as two keys are involved in encryption and decoding
Quantity of keys	One key is designed for encryption and decryption both in symmetric encryption	Two keys are designed for encryption and decryption in asymmetric encryption
Techniques	It is a time-tested method.	It uses a contemporary encryption method.
Confidentiality	There is a danger that a single key used for encryption and decoding will be compromised	There is no need to share a key thanks to the creation of two independent keys for encryption and decryption
Speed	Symmetric encryption is fast technique	Asymmetric encryption is slower in terms of speed.
Algorithms	RC4, AES, DES, 3DES, and QUAD.	RSA, Diffie-Hellman, ECC algorithms.

Li et al.'s In this work, Junhan YANG and Tianjie CAO demonstrated that the protocol was vulnerable to man-in-the-middle and off-line dictionary attacks. Furthermore, we have shown that the protocol proposed by Yoon et al. [10] still lacks backward secrecy and is vulnerable to off-line dictionary assaults. In addition, we have proposed an off-line elliptic curve verifier-based password authenticated key exchange protocol that is immune to dictionary attacks and server penetration. The suggested protocol can offer backward and forward secrecy as well as mutual authentication.

Two clients can create a shared session key based on their passwords using the Client-to-Client Password-Authenticated Key Exchange (C2C-PAKE) protocols. No opponent with constrained computational capabilities can learn anything about the shared session keys between two clients in a secure C2C-PAKE protocol. Session keys shouldn't be made available to the general public, especially to a participating server. Another crucial security feature is the C2C-PAKE protocol's resilience to impersonation brought on by server intrusion. This suggests that even if a client's password verifier is hacked, an external adversary should be unable to exchange the session key with any client. Recently, four C2C-PAKE protocols in the three-party environment were created by Kwon and Lee [11], and a cross-realm C2C-PAKE protocol was proposed by Zhu et al. [12]. Each suggested approach is said to prevent server compromise. Contrarily, we show in this study that server compromise attacks are possible against the

Kwon and Lee [11] and Zhu et al. [12] protocols and that a malicious server is capable of man-in-the-middle attacks that allow it to listen in on the discussion between the two clients.

The majority of the current password-based genuine key exchange techniques have proofs in either the Boyko, MacKenzie, and Patel (BMP) simulation-based security model or the Bellare, Pointcheval, and Rogaway (BPR) indistinguishable ability-based security model [13]. Although these models provide a level of security that is sufficient for the majority of applications, they ignore some realistic possibilities, such as participants using different but potentially linked passwords to execute the protocol. These issues were addressed by Canetti et alnew.'s security model under the universal composability (UC) framework, which makes no assumptions about the distribution of passwords used by protocol participants. A brand-new protocol was also proposed, but alas, it is not as efficient as some of the BPR and BMP procedures now in use.

Michel Abdalla, Dario Catalano, Celine Chevalier, and David Pointcheval, the paper's authors, investigate if some of the well-known procedures that have been shown to be secure in the BPR and BMP models can also be shown to be secure in the new UC model. We answer this question in the positive. More specifically, we show that the new UC model also supports the security of the Bresson, Chevassut, and Pointcheval (BCP) protocol from CCS 2003. The proof of security, which works even in the presence of adaptive adversaries capable of learning the internal states of players and corrupting them at any time, is built on the random-oracle and ideal-cipher models.

III. OVERVIEW OF TRNG

True Random Number Generators

The purpose of a TRNG (True Random Number Generator) is to produce non-deterministic data (for example, a series of numbers) to seed security algorithms. It is based on an unexpected physical occurrence known as an entropy source.

As connected gadgets enter our daily lives, we expect them to function well while safeguarding our personal and financial data. Given that secrets and other sensitive information are created and protected via these devices, TRNGs are the fundamental component of their security. They are a link in a "chain of trust" that must be built, beginning with the SoC and extending to the application layers and cloud communication. The strength of a trust chain is determined by its weakest link. [15]

RNGs that are predictable leave the door open to a wide range of potential hacking and data-compromise threats. Random numbers must be secure, uniformly distributed, unpredictable, statistically independent (unrelated to any previously generated random numbers), and safeguarded in order to be valuable.

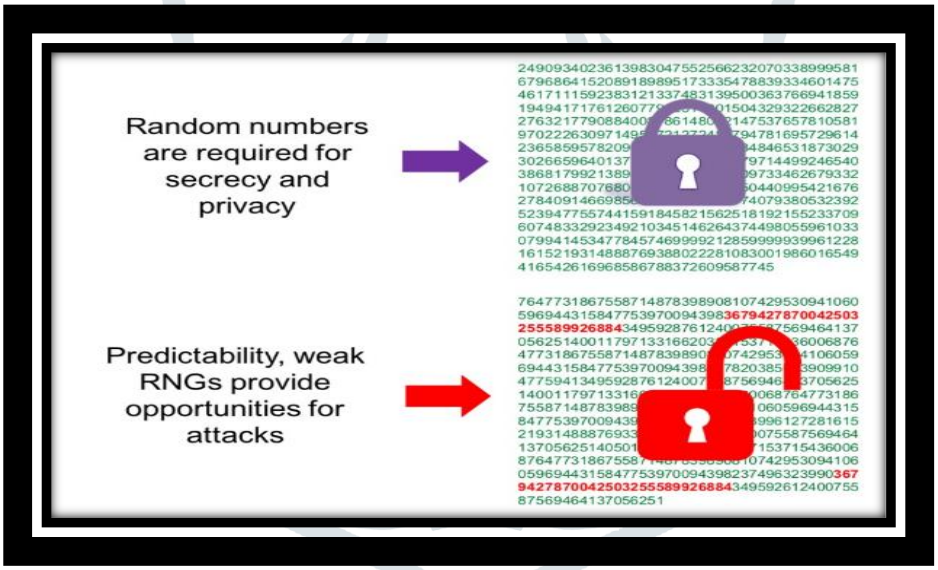

Figure 2True random numbers are crucial for security

RNGs that are predictable leave the door open to a wide range of potential hacking and data-compromise threats. For random numbers to be useful, they must be protected, statistically independent (unrelated to any previously created random numbers), uniformly distributed, and unpredictable.

The importance of true random numbers and physical nondeterministic random number generators (RNGs) seems to be growing. Mathematics, stochastic and quantum cryptography, Monte Carlo simulations, statistical analysis, randomised algorithms, lottery, and other applications all depend on random numbers. The numerous applications of cryptography to our daily lives, including mobile communications, email access, online payments, cashless payments, ATMs, e-banking, internet trade, point of sale, prepaid cards, wireless keys, general cyber-security, distributed power grid security (SCADA), etc., make true random numbers increasingly necessary today. We shall assume that generators produce random bits for the remainder of the text to maintain generality.

Due to the Kerchoff's principle, a random number generator that is appropriate for cryptography must create completely unpredictable bits even if its entire design (schematic, algorithms, etc.) is known. Physical (true, hardware) random number generators, as opposed to PRNGs, get randomness from fundamentally nondeterministic physical processes, making them superior candidates for real random number generation. A physical RNG is an independent piece of hardware that is typically connected to the computer through USB or PCI connection. It is difficult and necessitates original drivers to import random integers into a user programme.

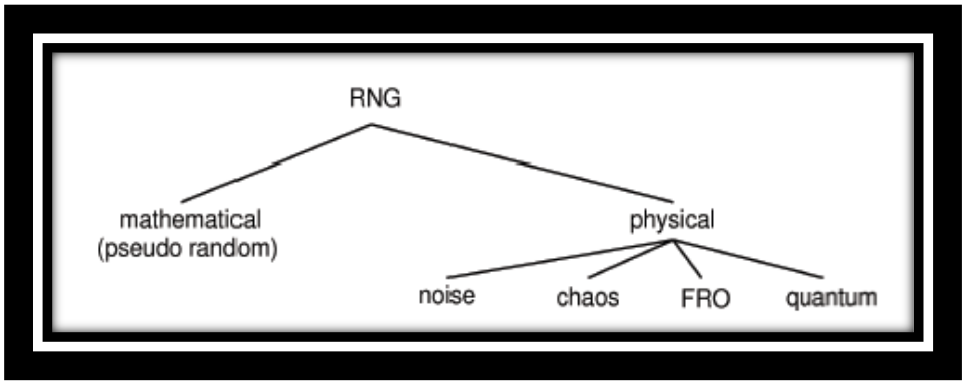

Figure 3Classification of random number generators.

Pseudorandom Number Generators

There have historically been two methods of generating random numbers: algorithmically (pseudorandom) and by a physical process (nondeterministic). We won't go into great depth here because pseudorandom number generators (PRNG) are well known in the field. You can find surveys and specific examples of PRNGs elsewhere [15]. In essence, a PRNG is just a mathematical formula that generates a deterministic, periodic sequence of numbers that is entirely governed by the beginning condition known as seed. Such generators are by definition not deterministically random. However, they also have significant long-range correlations that threaten the security of cryptography and can result in unanticipated errors in Monte Carlo calculations and modelling. In actuality, PRNGs have 0% bias and the ideal balance of 0s and 1s. Even while the bulk of modern PRNGs pass all known statistical criteria, there are misunderstandings that some PRNGs are significantly better than others. The fact is that every PRNG reveals its flaws in a specific application. In fact, PRNGs are frequently identified as the root of inaccurate stochastic simulations and calculations [16]. Since all significant families of PRNGs have been crypt analyzed for cryptographic purposes [17], using a PRNG when a RNG should be used will greatly increase the security risk for the protocol in issue.

IV. CONCLUSION

Since the seed value utilised in the random number generator comes from an unreliable source, the results are truly random. As a result, it entirely eliminates the risk of determinism, which is present in pseudo-random number generators.

It is also substantially easier to design than alternative hardware random number generators that provide randomness utilising radioactive decay, atmospheric noise, electrical noise, etc. By combining data from fish tank photos with the TRNG, truly random numbers can be generated at a reasonable cost. Synopsys provides a wide range of fully integrated security IP solutions for a number of devices in the mobile, automotive, digital home, IoT, and cloud computing markets in addition to TRNGs. These solutions rely on a common set of security ideas and standards-based building components.

REFERENCES

1. S. He, W. Zeng, K. Xie, H. Yang, M. Lai, and X. Su, "PPNC: privacy preserving scheme for random linear network coding in smart grid," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 3, pp. 1510–1533, 2017.
2. K. Gu, N. Wu, B. Yin, *and* W. Jia, "Secure data sequence query framework based on multiple fogs," *IEEE Transactions on Emerging Topics in Computing*, 2019. [8] K. Gu, K. Wang, and L. Yang, "Traceable attribute-based signature," *Journal of Information Security and Applications*, vol. 49, pp. 1–13, 2019.
3. K. Gu, X. Dong, and L. Wang, "Efficient traceable ring signature scheme without pairings," *Advances in Mathematics of Communications*, 2019.
4. G. Cheng, C. Wang, and H. Chen, "A novel color image encryption algorithm based on hyperchaotic system and permutation-diffusion architecture," International Journal of Bifurcation and Chaos, vol. 29, no. 9, p. 1950115, 2019.
5. M. Long, F. Peng, and H. Y. Li, "Separable reversible data hiding and encryption for HEVC video," Journal of Real-Time Image Processing, vol. 14, no. 1, pp. 171–182, 2018.
6. A. A. Rezk, A. H. Madian, A. G. Radwan, and A. M. Soliman, "Reconfigurable chaotic pseudo random number generator based on FPGA," *AEU-International Journal.*
7. Hugo Krawczyk. SIGMA: The \SIGn-and-MAc" approach to authenticated Diffe-Hellman and its use in the IKE protocols. In Dan Boneh, editor, Advances in Cryptology { RYPTO 2003, volume 2729 of Lecture Notes in Computer Science, pages 400{425, Santa Barbara, CA, USA, August 17{21, 2003. Springer-Verlag, Berlin, Germany.
8. Steven M. Bellovin and Michael Merritt. Encrypted key exchange: Password-based protocols secure against dictionary attacks. In 1992 IEEE Symposium on Security and Privacy, pages 72{84, Oakland, CA, May 1992. IEEE Computer Society Press.
9. S.-W. Lee, H.-S. Kim and K.-Y. Yoo, Efficient verifier-based key agreement protocol for three parties without server's public key, Applied Mathematics and Computation, 167(2), pp. 996-1003, 2005.
10. R-C Wang and K-R Mo, Security enhancement on efficient verifier-based key agreement protocol for three parties without server's public key, Int. Math. Forum, 1(17-20), pp. 965 – 972, 2006.
11. E.J. Yoon, and K.Y. Yoo. Robust User Password Change Scheme based on the Elliptic Curve Cryptosystem. In FundamentaInformaticae, pages 483-492, 2008.

Zuowen Tan "An Enhanced Three-Party Authentication Key Exchange Protocol for Mobile Commerce Environments", JOURNAL OF COMMUNICATIONS, VOL. 5, NO. 5, pp. 436-443, MAY 2010.

12. G. Yang, D. S. Wong, H. Wang, X. Deng, "Two-factor mutual authentication based on smart cards and passwords", Journal of Computer and System Sciences, Vol. 74, No. 7, pp.1160-1172, November 2008.
13. Michel Abdalla, Dario Catalano, Celine Chevalier, and David Pointcheval "Efficient Two-Party Password-Based Key Exchange Protocols in the UC framework", Proceedings of the Cryptographers' Track at the RSA Conference (CT-RSA '08), pages 335{351, Springer-Verlag, 2008.
14. MihirBellare, David Pointcheval, and Phillip Rogaway. Authenticated key exchange secure against dictionary attacks. In EUROCRYPT 2000, LNCS 1807, pp. 139-155. Springer-Verlag, Berlin, Germany, May 2000.
15. D. E. Knuth. High speed single photon detection in the near infrared. The Art of Computer Programming, Vol. 2, 3rd Edition, Addison Wesley, 1997.
16. O. Kwon, YQuantum random number generator using photon-number path entanglement. Appl. Optics, 48:1774-1778, 2009.
17. G. Parisi and F. Rapuano. Effects of the random number generator on computer simulations. Physics Letters B, 157:301-302, 1985.