An Approach to Generate Automatic Variable Key to Assure Perfect Security in Cryptosystem

Subhasish Banerjee, Manash P. Dutta and C.T. Bhunia

Abstract Due to advancement and worldwide rapid deployment of computer networks, the security becomes a major issue. Many mechanisms have been devised and more new ideas are about to propose in coming era. But, all the existing techniques have only one important criterion that key must be sustained and protected at any circumstances. In this regard, the AVK is one of the novel and unbreakable approaches to fulfill such criteria as being experimented so far. In this paper, we have proposed a new technique to generate AVKs which can provide higher security by enhancing the randomness among the generated successive keys. To adhere to our claimed, the comparative studies with existing techniques have also been cited at the end.

Keywords Common divisor · Variable key · Randomness · Security

1 Introduction

As computer networks becomes an essential and important part of our daily lives, protecting information and information system from unauthorized access, disruption, modification, use, disclosure or destruction becomes a complicated and challenging task among the researchers. From last few decades, after putting rigorous effort, many new schemes and ideas have been proposed to fulfill such requirements. But, due to the advancement and growth of computer technology day by day, attaining the security becomes an abundant issue with a single key. In this regard, Shannon [1, 2] devised that if key changes from session to session or time to time or make the size of key as same as that of plain text then to break the code with cipher text only attack becomes infeasible. Herein, lies the necessity of Automatic Variable Key (AVK) [3–5]. AVK makes the key dynamic in nature rather than static one throughout the transfer of data communication between sender and receiver. AVK mechanism

S. Banerjee(✉) · P. Dutta · C.T. Bhunia
Department of CSE, National Institute of Technology, Yupia 791112, Arunachal Pradesh, India
e-mail: subhasishism@gmail.com, manashpdutta@gmail.com, ctbhunia@vsnl.com

S. Latifi (ed.), *Information Technology New Generations*,
Advances in Intelligent Systems and Computing 448,
DOI: 10.1007/978-3-319-32467-8_4

generates the keys in such a way that key changes every time whenever a new block of data is exchanged. The main key features of AVK is that due to random variation among the successive keys the probability of the cipher text attacks such as frequency attack, differential frequency attack, brute force attack etc. has been reduced tremendously according to theoretical survey by the researchers. Thereafter, many researchers like Chakraborty et al. [6], Goswami et al. [7, 8], Banerjee et al. [9–11] and Dutta et al. [12–14] had proposed their ideas and gave their contributions towards the security over insecure communication channel. In this paper we have proposed a new technique for key generation which can upgrade the level of security by increasing the randomness among the successive keys. The rest of the paper has been organized as: proposed scheme with key generation examples have been included in section 2 and 3 respectively. In section 4, we have summarized our experimental results. To prove its excellency we have compared our scheme with the existing schemes in section 5. Finally, we have concluded in section 6.

2 Proposed Scheme

In this section, we have described our proposed technique namely Automatic Variable Key based on Common Divisor (AVKCD) to generate the automatic variable keys. In this technique, the keys used to change based on common divisor of greatest and the smallest (Other than One) between the previous key and previous block of data. The explanation of our proposed algorithm is mentioned in the following way:

```
$Key\ Generation(Initial\ Key, Data\ set)$
{
        $K_1 = Initial\ Key$
        $i \leftarrow 1$
        $repeat$
        {
                $R \leftarrow GCD(K_{i-1}, D_{i-1})$
                $S \leftarrow SCD(K_{i-1}, D_{i-1})$
                $if(S == 0)then$
                $S \leftarrow 1$
                $K_i = CRS(K_{i-1}, R) \oplus CLS(D_{i-1}, S)$
                $i++$
        }$until(Data\ set \neq \emptyset)$
}
```

Where:
$CLS(x, y)$ = Circular Left Shift of x by y bit positions.
$CRS(x, y)$ = Circular Right Shift of x by y bit positions.
$GCD(x, y)$ = Greatest Common Divisor of x and y
$SCD(x, y)$ = Smallest Common Divisor of x and y other than one

3 Thoretical Example of Key Generation

In this section, we have summarized our proposed technique by defining some key generation examples. In this scheme, the key generator starts the operation to produce the successive keys after the establishment of initial key between the sender and receiver. For simplification, we have considered the block size of plain text and initial key are 8 bits only. Here, we have assumed the initial key K_1 as 10001010 and initial data D_1 as 10101010. The necessary steps for generating the successive keys are given as below:-

Step 1 The GCD and SCD of decimal equivalent of $K_1(=138)$ and $D_1(=170)$ are 2 and 2 respectively. Therefore, 2^{nd} key, K_2 can be calculated by $CRS(10001010, 2) \oplus CLS(101010, 2) = 10100010 \oplus 1010100 = 00001000$.

Step 2 To compute 3^{rd} key K_3, assumed 2^{nd} data block D_2 was 11011000, hence GCD and SCD of decimal equivalent of $K_2(=8)$ and $D_2(=216)$ are 8 and 2. Henceforth, K_3 is $CRS(00001000, 8) \oplus CLS(11011000, 2) = 00001000 \oplus 01100011 = 01101011$.

Step 3 To yield the next key K_4, 3^{rd} data block D_3 was assumed as 11100100, therefore GCD and SCD of decimal equivalent of $K_3(=107)$ and $D_3(=228)$ are found as 1 and 0 respectively. But, as per the algorithm, if SCD is 0 then it is assigned as 1. Hence, the computed value of K_4 is $CRS(01101011, 1) \oplus CLS(11100100, 1) = 10110101 \oplus 11001001 = 01111100$.

In this way, successive keys can be generated based on the availability of data set.

4 Experimental Results

In this section, we have demonstrated various experiments to show the efficiency of our proposed scheme. In all the following experiments, we have assumed that the length of all the individual datasets is 8 bits long. For simplification, we have taken 8 bit initial key $K_1 = 10001010$. While plotting the graph, we have considered first 100 keys only.

Experiment 1:-The dataset for this experiment is taken as *A brute-force attack involves trying every possible key until an intelligible translation of the cipher text into plaintext is obtained.* The generated successive keys shows the randomness which are depicted in Fig. 1.

Experiment 2:-Here, we have taken *Frequency analysis is based on the fact that, in any given stretch of written language, certain letters and combinations of letters occur with varying frequencies.* as data set to get the randomness and the graph is given in Fig. 2.

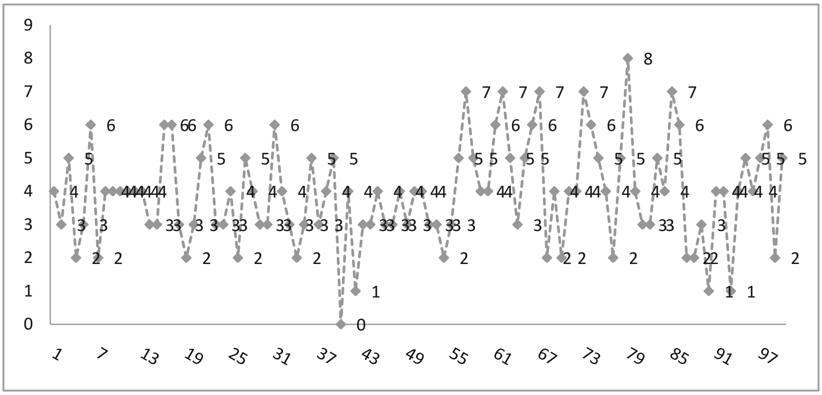

Fig. 1 Randomness of the successive keys

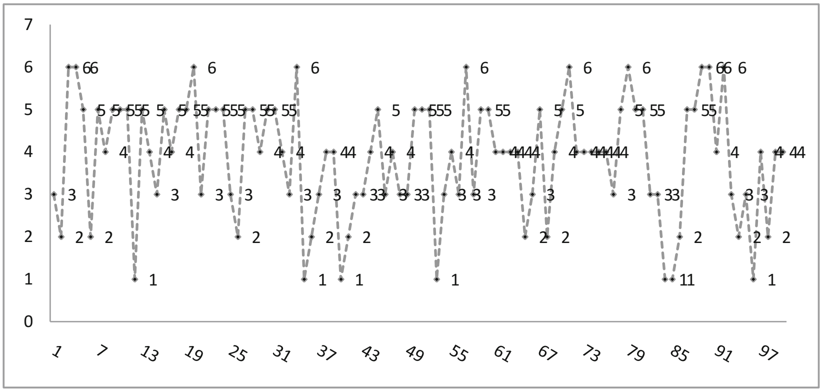

Fig. 2 Randomness of the successive keys

Experiment 3:-In the last experiment, the data set are taken as *An encryption scheme is said to be computationally secure if either of the foregoing two criteria are met.* to calculate the randomness of the auto generated successive keys and the generated graph is included in Fig. 3.

5 Performance Comparison

In this section, we have compared the efficiency of our proposed scheme by comparing the average randomness with the other related schemes. The average randomness has been calculated as the same fashion which has been described in the

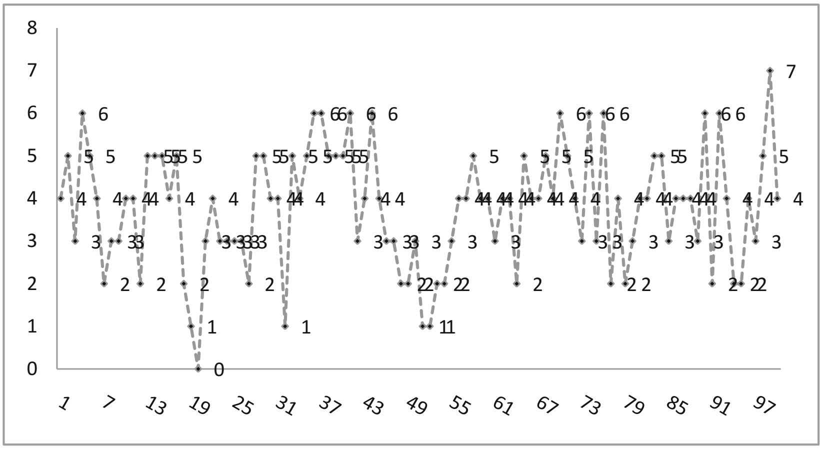

Fig. 3 Randomness of the successive keys

compared schemes. As because, the randomness among the keys is highly dependent on the initial key and data set pairs, we have taken all the above experiments into our account. The average randomness comparison graphs for the experiments 1 to 3 have been plotted in Fig. 4.

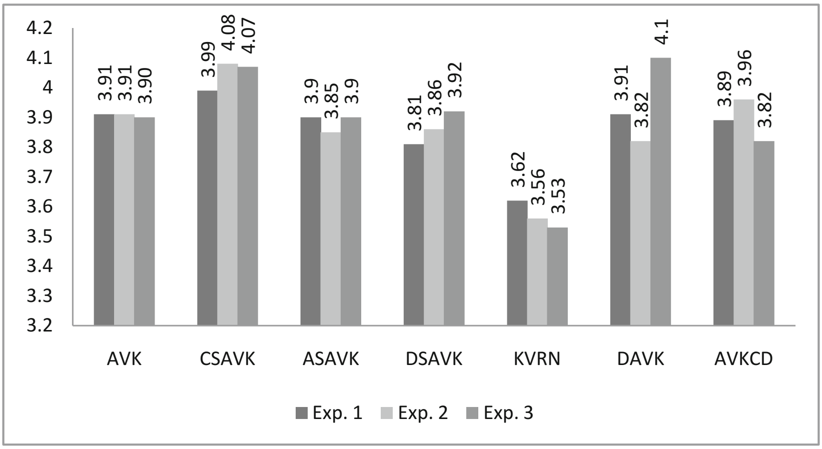

Fig. 4 Average Randomness Comparison with the Various Existing Techniques

To define the actual degree of heterogeneity, we have also compared our scheme using experiments 1 to 3 based on standard deviation as well. The corresponding graphs have been plotted from Figure 5 to 7.

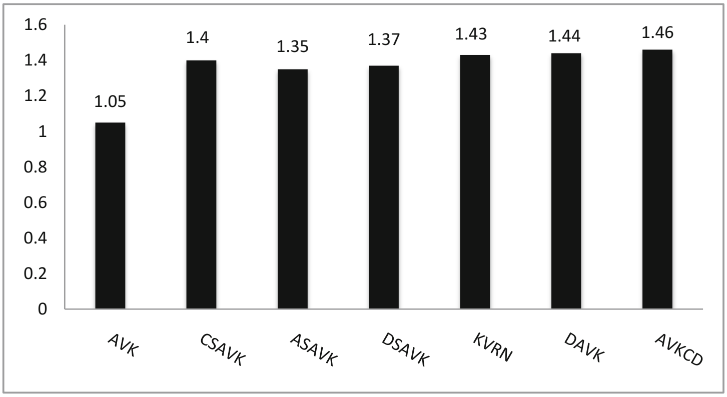

Fig. 5 Standard deviations comparisons for the experiment 1

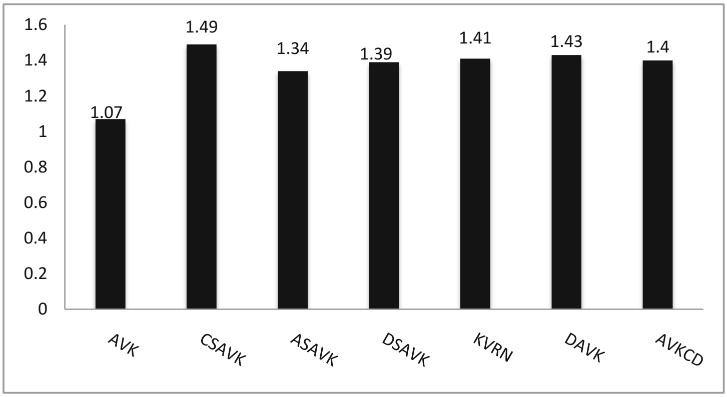

Fig. 6 Standard deviations comparisons for the experiment 2

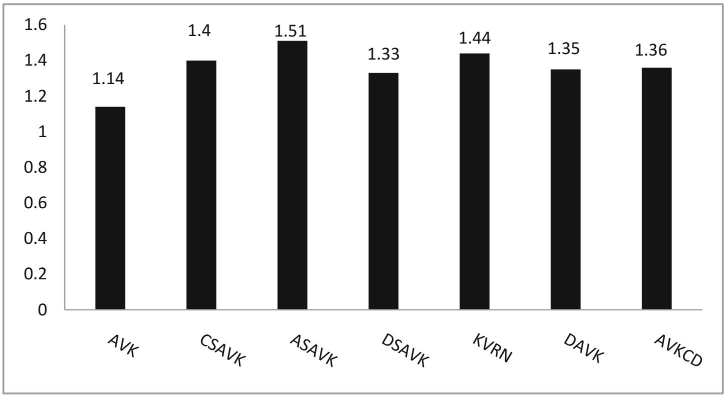

Fig. 7 Standard deviations comparisons for the experiment 3

6 Conclusion

In this paper, we have proposed a new mechanism to generate the AVKs and named AVKCD. Based on the comparisons performed with various related experiments given under performance analysis section, it can be stated that our technique is superior to other techniques in terms of standard deviation as it is the best way to define the actual degree of heterogeneity than average randomness. Therefore, it can be concluded that our proposed scheme can enhance the security by changing the keys from session to session during data communication.

References

1. Shannon, C.E.: Mathematical theory of communication. Bell System Technical Journal **27**(379–423), 623–656 (1984)
2. Shannon, C.E.: Communication theory of secrecy system. Bell System Technical Journal **28**, 656–715 (1949)
3. Bhunia, T.C.: New approaches for selective AES towards tackling error propagation effect of AES. Asian Journal of Information Technology, 1017–1022(2006)
4. Bhunia, T.C.: Information technologynetwork and internet, vol. 1. New Age Publication (2005)
5. Bhunia, T.C., Mondal, G., Samaddar, S.: Theories and application of time variant key in RSA and that with selective encryption in AES. Proc. EAIT, 219–221 (2006)
6. Chakraborti, P., Bhuyan, B., Chowdhuri, A., Bhunia, C.T.: A novel approach towards realizing optimum data transfer and automatic variable key (AVK). International Journal of Computer Science Network Security **8**(5), 241–250 (2008)
7. Goswami, R.S., Chakraborty, S.K., Bhunia, A., Bhunia, C.T.: Generation of Automatic Variable Key under various Approaches in Cryptography System. Journal of Institute Engineering India Series B **94**(4), 215–220 (2014)
8. Goswami, R.S., Chakraborty, S.K., Bhunia, A., Bhunia, C.T.: Approach towards Optimum Data Transfer with various automatic variable key techniques to achieve perfect security with analysis and comparison. International Journal of Computer Applications **82**(1), 28–32 (2013)
9. Banerjee, S., Dutta, M.P., Bhunia, C.T.: A Novel Approach to Achieve the perfect security through AVK over Insecure Communication Channel. Journal of Institute Engineering India Series B (Communicated)
10. Banerjee, S., Dutta, M.P., Bhunia, C.T.: A New Three Dimensional Based Key Generation Technique in AVK. Journal of Institute Engineering India Series B (Communicated)
11. Singh, B.K., Banerjee, S., Dutta, M.P., Bhunia, C.T.: Generation of automatic variable key to make secure communication. In: International Conference on Recent Cognizance in Wireless Communication and Image Processing-ICRCWIP-2014 (2015)
12. Dutta, M.P., Banerjee, S., Bhunia, C.T.: An Approach to Generate 2-Dimensional AVK to Enhance Security of shared Information. International Journal of Security and Its Applications **9**(10), 147–154 (2015)
13. Dutta, M.P., Banerjee, S., Bhunia, C.T.: Two new schemes to generate automatic variable key (avk) to achieve the perfect security in insecure communication channel. In: Proceedings of the International Conference on Advanced Research in Computer Science Engineering and Technology (ICARCSET, Eluru, India) (2015)
14. Dutta, M.P., Banerjee, S., Bhunia, C.T.: Generation of variable session keys based on piggybacking strategy. In: Proceedings of 3rd International Conference on Advances in Computing, Electronics and Communication (Zurich, Switzerland) (2015)