The International Journal of Computer Science and Communication Security (IJCSCS) Vol. 6, Feb 2016

Reduction of Encryption Key Search Space Based on The Min-Entropy Approach

Magdy Saeb,

Arab Academy of Science, Technology and Maritime Transport, Alexandria, Egypt

Great Wall Information Security (GWIS), Kuala Lumpur,

mail@magdysaeb.net

Abstract: Encryption key search is basically guessing the value of a random variable X by trials of the form "Is X equal to x?" until the correct value is found. We show how to reduce the search space from 2^n for a brute force attack or $2^{n/2}$ for a Birth Day Attack to a smaller number of trials less $2^{n/2}$ where n is the number of key bits. This is accomplished using the fact that the internally generated keys, by design, have a ratio of the number of ones approximately ranging from 0.45 to 0.499 to maximize the key entropy H. Using this simple observation, the search range is reduced from O ($2^{n/2}$) to O (2^{Δ}), where Δ is the range specified by min-entropy for key guessing with added available information to the adversary.

Key words: key, search, cipher, min-entropy, search space, key guessing.

1. Introduction

Encryption key search is basically guessing the value of a random variable x by trials of the form "Is X equal to x?" until the correct value is found. The problem of key search was studied by Massey [Mas94]. Cryptanalysis of computationally secure ciphers more or less depends on performing these trials. For a computationally-secure cryptosystem, a related-key attack can be launched for finding the secret key. In this type of attacks, the adversary observes the operation of the cipher under several different keys. These keys have some mathematical relation connecting them that is possibly known to the adversary. This type of attacks is especially relevant for symmetric ciphers. Other techniques, like linear [YosMis97] or differential cryptanalysis [BiSha91] can be used to reduce the number of keys that must be tried [MvOV97].

In This work, we show how to reduce the search space from 2^n for a Brute Force Attack or $2^{n/2}$ for a Birth Day Attack (BD attack) to 2^{Δ} where Δ is < n/2. This is accomplished using the fact that internally generated keys, by design, have a ratio of the number of zeroes or ones approximately ranging from 0.45 to 0.50 in order to maximize the key entropy. Using this simple observation, the search range is reduced from O ($2^{n/2}$) to O (2^{Δ}). Regarding the time complexity, the form 2^n is called sub-exponential form. We call our search range "The Delta Search". One compares the range from 0.0 to 0.5 for BD attack to the range from 0.45 to 0.5 for the delta search (0.5-0.45)/0.5-0) = 0.05/0.5 = 0.1. That is about 90% reduction in search effort. The observation serves as the partial information given to the adversary to guess the key correctly.

In the following few sections of this short correspondence, we discuss the concept min-entropy as applied to key guessing. We show that ciphers, by design, generate and use key with the ratio of number of ones to the total key size ranges from 0.45 to a value that is slightly less than 0.5 in order to maximize the entropy of the round keys and remain in the stable region as indicated by the relation of the entropy versus the probability of number of ones in the round key. Sometimes this is called Hamming Weight. In this case the Hamming

weight is equal to the Hamming distance. We propose a simple search procedure to find the encryption key in this relatively smaller search space. Finally, we give a summary and our conclusions.

2. Key Guessing and The Min-Entropy

The probability that the correct value is guessed in the first trial is directly linked to the min-entropy of X and is equal to:

$$\text{Min-entropy} 2^{-H_\infty(X)} = \max_{x \varepsilon X} P_X(x) \quad (1)$$

In other words, a random variable X has min-entropy k, denoted by:

$$H_\infty(X) = k, \text{ if } \max_x \Pr\{X = x\} = 2^{-k} \quad (2)$$

That is,

$$H_\infty(X) = -\log_2 \max_x \Pr\{X = x\} \quad (3)$$

If we call the condition X= x, W, where W is the action of the adversary predicting the sample correctly. Then

$$H_\infty(X) = -\log_2 \max \Pr\{W\} \quad (4)$$

That is, min-entropy is equal to -log Pr {the adversary predicts sample correctly}, [Cac97], [Reyz11], [AnMa13].

Figure 1 demonstrates the search spaces with and without an initial guess. The dark highlighted area shows the effect of reducing the search space from n/2 to Δ.

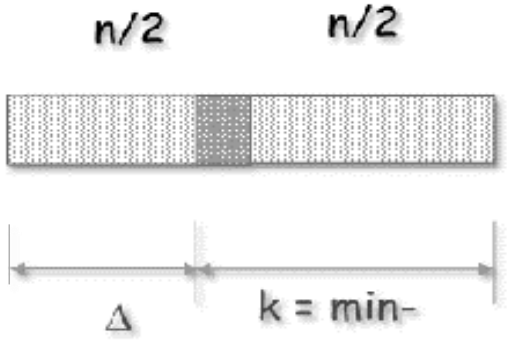

Figure 1. Guessing the key by reducing the search space to Δ

Now the inquiry that remains here is what is the optimal guess to reduce the key search space? Referring to Figure 2, one immediately observes that the round keys are situated in the left side of the curve representing the relation between the entropy H versus the probability of the number of ones in the key bits. The region, as shown in Figure, has a stable (the left side), critically stable (max entropy) and unstable regions (the right side).

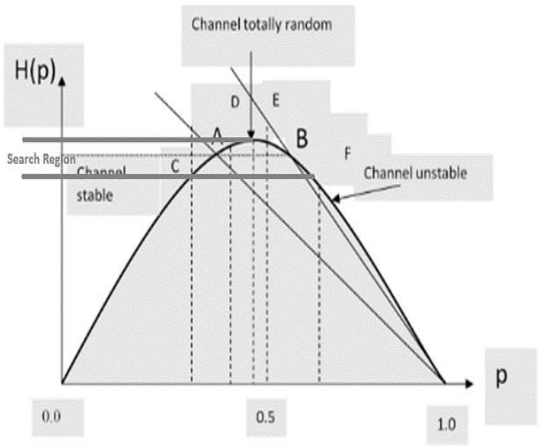

Figure 2. The Entropy H(p) versus number of changed bits' plot indicating stable and unstable channel regions of operation

3. The Most Probable Key Locations

When we lose our house key and try to find it, we usually start with the most probable locations then gradually expand the search domain until the key is found or we abandon the search. However, in encryption key search not applying linear and differential key attacks, we tend to do the opposite by starting with brute force using state-of-the-art short cycle time computers. As mentioned before, and using the fact that the internally generated keys or round keys have the number of ones around a ratio that ranges between the end and start values equal to $R_e = 0.45$ to $R_s = 0.50$, the search space can be

tremendously reduced. If the key is not found in this range, we expand the search domain by reducing R_e in steps. This procedure is shown in Figure 3 using Nassi-Schneiderman Chart.

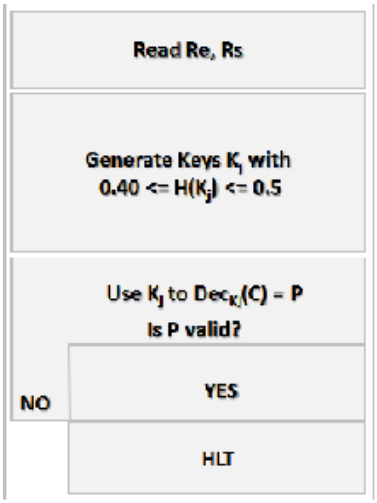

Figure 3. The search procedure based on the most probable correct key location

This procedure is summarized as follows:

Procedure: Reduced-Key-Search-Space

Input: Rs, Re, delta

Output: k ε K

Begin

1. Read Rs, Re;
2. Generate Keys K_j with
   Re <= H(K_j) <= Rs
3. While P = NOT VALID

   Dec_{Kj} (C) = P;

   Next K_j
4. If key found, then Output K_j

   Else

   {Re = Rs,

   decrement Rs = Rs – delta,

   GOTO 2};

   End
5. HLT.

As an example we use a standard hash function such as SHA-512 to hash, say, a user key "HelloHello". The result can be thought of as the initial round key. The output of the hash function is given by:

fc6996c8c2e7bc7bd47510d33b0aa4cd0f214e0
ddd12fe04b6333d5ba452504e379cba523113d
9e9f888c25c3f488ce2c9d7e007e4a0f9c46cf0f
b51c089e19f

The Binary value of this hexadecimal is given by:

1111 1100 0110 1001 1001 0110 1100 1000
1100 0010 1110 0111 1011 1100 0111 1011
1101 0100 0111 0101 0001 0000 1101 0011
0011 1011 0000 1010 1010 0100 1100 1101
0000 1111 0010 0001 0100 1110 0000 1101
1101 1101 0001 0010 1111 1110 0000 0100
1011 0110 0011 0011 0011 1101 0101 1011
1010 0100 0101 0010 0101 0000 0100 1110
0011 0111 1001 1100 1011 1010 0101 0010
0011 0001 0001 0011 1101 1001 1110 1001
1111 1000 1000 1000 1100 0010 0101 1100
0011 1111 0100 1000 1000 1100 1110 0010
1100 1001 1101 0111 1110 0000 0000 0111
1110 0100 1010 0000 1111 1001 1100 0100
0110 1100 1111 0000 1111 1011 0101 0001
1100 0000 1000 1001 1110 0001 1001 1111

One immediately notices that the number of ones in this output is equal to 253 with a ratio of 253/512 = 0.49414. One can try other inputs to this or any other hash function or cipher and more or less he or she will get approximately similar results. As a matter of fact, the expected output of all hash functions, and ciphers in this respect, will follow the same trend. Therefore, we should concentrate the key search in this region. Now, one may instantly inquire about the security of existing ciphers. The answer is simple. We have to increase the key sizes to 512 or even 1024 bit or more.

Summary and Conclusions

We have shown that the possible decryption keys are not scattered all over the range of values with entropy 0 to 1 but rather are intrinsically crammed in a small range of entropy. The reason behind this, is the design of hash functions and ciphers to maximize their entropy content. Using this fact, the search process for round keys is reduced by a factor of about 90% as compared to BD attacks. This approach capitalizes on the fact that the round keys compose a small set as compared to the set anticipated by a brute force attack. We consider this observation to be the required bestowed information later acquired by the adversary and represented by the min-entropy formulation of the key guessing problem.

References

[Mas94] James L. Massey, "Guessing and entropy," Proceedings of the 1994 IEEE International Symposium on Information Theory, page 204, 1994.

[MvOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, Handbook of applied cryptography, CRC Press, Boca Raton, FL, 1997.

[BiSha91] E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Journal of Cryptology, vol. 4, no. 1, pp. 3-72, 1991.

[YosMis97] A.M. Youssef, S. Mister, and S.E. Tavares, "On the Design of Linear Transformations for Substitution Permutation Encryption Networks", Workshop on Selected Areas of Cryptography (SAC '96): Workshop Record, pp. 40-48, 1997.

[Cac97] Christian Cachin, Entropy Measures and Unconditional Security in Cryptography, a doctoral dissertation submitted to the Swiss Federal Institute of Technology, Zurich, 1997.

[Reyz11] Leonid Reyzin, "Some Notions of Entropy for Cryptography," Boston University Computer Science http://www.cs.bu.edu/~reyzin, June 2, 2011.

[AnMa13] Lu´ıs Antunes, Armando Matos, Alexandre Pinto, Andreia Teixeira, "One-way functions using Algorithmic and Classical Information Theories," Instituto de Telecomunica¸ andreiasofia@dcc.fc.up.pt, October 28, 2013.

Magdy Saeb received the BSEE. School of Engineering, Cairo University, in 1974, the MSEE, and Ph.D. Degrees in Electrical & Computer Engineering, University of California, Irvine, in 1981 and 1985, respectively. He was with Kaiser Aerospace and Electronics, Irvine California, and The Atomic Energy Establishment, Anshas, Egypt. He is a professor emeritus in the Department of Computer Engineering, Arab Academy of Science, Technology & Maritime Transport, Alexandria, Egypt; He was on leave working as a principal researcher in the Malaysian Institute of Microelectronic Systems (MIMOS). Now he is the CTO of Great Wall Information Security, Kuala Lumpur, Malaysia. His current research interests include Cryptography, FPGA Implementations of Cryptography and Steganography Algorithms, Encryption Processors, Mobile Agent Security.

www.magdysaeb.net