Extending Shannon Security to Variable Size Keys of No Preset Limit

Gideon Samid
D&G Sciences – Virginia Technology Corporation
P.O.Box 1022 McLean VA 22101-1022
www.dgsciences.com
Gideon@dgsciences.com
703.385.4144

"Shannon Security" is attainable for stream ciphers where the key is as long as the message. One asks: what if the key is of variable (secret) size without a preset high limit? Formally this case is a generalization of the former, but it also allows for Shannon Security to apply for smaller key length, which is where it becomes interesting. We explore the circumstances where one may achieve "Meta Shannon Security" without the burden of a message-size key.

Introduction

Cryptography today is haunted by the specter of accelerated brute force attack. In a typical case the exposed ciphertext is doomed to eventually yield its hidden secret. Users simply hope that it would take long enough for a persistent cryptanalyst to dig out the buried plaintext. Such hope is well founded against hackers and code crackers who are not as clever, or not smarter than the cryptographer who designed the system. In other words, modern cryptography bets *against* innovation; it discounts the prospect of an ingenious shortcut that would violate its prized security. But doubts linger.

The one remedy for this concern is known as Shannon Security. This is a situation where the ciphertext will not betray its trusted secret even in the face of overwhelming computing power. More dramatically: A Shannon Security ciphersystem is immunized against future mathematical insight. Its principle: ***equivocation defeats cryptanalysis***. The ciphertext may be linked to several plausible plaintexts which are cryptographically equivalent, but only one of them is the true plaintext.

While a Shannon security algorithm was invented in 1917 (The Vernam Cipher, or One Time Pad -- OTP), its use was rather limited since it requires a key as long as the encrypted message. It was widely believed that this is the lower limit for the required key size.

In this paper we seek to explore the possibility of extending Shannon security (albeit, with some sacrifice) into smaller size keys. Such extension can be accomplished through variable size keys of no preset limit.

Narrow Interpretation of Shannon Security

Shannon Security is defined as a situation where capturing the ciphertext is not at all helpful for the effort to retrieve the hidden plaintext. The familiar "One Time Pad" (OTP) cipher exhibits Shannon Security: any ciphertext, C, can be matched with same size plaintext, P, by XOR-ing it with a key, K which in turn, is computed as: $K = C \oplus P$

Strictly speaking the OTP ciphertext does give some information about the corresponding plaintext P (its size). And hence one might opt for the following -- call it narrow -- definition of Shannon Security:

A ciphersystem exhibits Shannon Security, if there exist n>0 plaintexts $P_0, P_1, P_2, P_3, ...P_n$. which are all above a given threshold, T, of circumstantial plausibility, and each may be associated with a corresponding key: $K_0, K_1, K_2, K_3, ...K_n$ such that:

$$C = E(P_i, K_i) \text{ for } i=0,1,2,....n$$

Where E is the encryption system.

We may further designate such a system as exhibiting Shannon Security of Order n.

Naturally, the higher the n value, the better the security. Even for n=1, the cryptanalyst will exhaust his or her analytic effort by identifying P_0 and P_1, without being able to further distinguish between them. Formally a Shannon Security of Order 0 is a "no Shannon Security." It means that for a given ciphertext C there exists only a single plaintext, P_0 which is both plausible and corresponds to C through some valid key K_0.

RSA, DES and all other prevailing ciphersystems exhibit zero Shannon Security in the general case. (In some rare particular circumstances a higher level of Shannon Security may be registered).

As a corollary we may use the notion of deniability. Shannon Security of order n amounts to cryptographic deniability of order n. Meaning: the user will be able to deny that message P_0 was encrypted into C, and claim that it was rather P_i. (i=1,2,...n).

In general the OTP comes with Shannon security of an extremely large order. (Based on the size of the message and its entropy). However, in a practical case the number (n+1) of plausible messages may be much smaller. Yet all the plausible (even non plausible) messages will easily be linked to the given ciphertext C via a corresponding key. That is the nature of One Time Pad.

The Advantage of Variable Size Key of No Preset Limit

The advantage of variable size key is the added key variability in comparison to a fixed size key. The cryptanalyst is burdened with an additional uncertainty which is not only quantitative but also qualitative. A cryptographic system where the key may grow beyond any preset limit can not be

brute-force exhausted. The latter is an important theoretical advantage but it may be reduced into practical insignificance because of the tedium and undue burden of using extremely large keys. The no-limit attribute should be used as a "teaser" option to confound the cryptanalyst, while in most real life cases, a very small key should be selected.

Below we discuss some key size aspects, and develop the notion of decoy keys and decoy plaintexts (deniability).

Small (False) Keys Are Easy to Use, but Hard to Find

The issue arises only with respect to a general case cryptography where the volume of messages encrypted with a single key is very large. (Variable key size of no set limit is readily useful for short messages).

The longer the message P (which can be referred to as the aggregate message encrypted by a single key), the more impractical it is to use a message-size key. Or say, the more likely is it that the user employed a key which is much smaller than the message.

On the other hand, the smaller the key, the more difficult it is to fit a given pair of ciphertext-plaintext with a matching key. As we see it with the prevailing fixed-size key cryptographies, an arbitrary random ciphertext string (if it's long enough) will not have a corresponding key (in the finite key space) which will generate a match with a plausible plaintext.

In particular with stream ciphers: it is quite easy to generate an endless pseudo-random number list from a small as desired seed. The reverse is where difficulties mount.

Say then that if a given ciphertext, C, can be matched with two pairs of plaintext-key:

$$C = E(P_1,K_1) = E(P_2,K_2)$$

Where E is the encryption function, and if one key is much smaller than the other:

$$K_1 << K_2$$

then it would be a "sound guess" to claim that P_1 is the true message which is hidden in C, while P_2 is a decoy message.

If E is a stream cipher, then one can use K_1 to encrypt P_1, and then point to an arbitrary message P_2 for which one would easily find K_2 as large as P ($K_2 = C \oplus P_2$), and mount a feeble, yet theoretical claim that it is mathematically unclear which is the true message and which the decoy.

The subject of decoy keys and decoy plaintexts is further elaborated below.

Decoy Keys; Decoy Messages

A cryptographic system that employs a key of variable size which, in turn, may become as large as necessary, is akin to message-size key system in as much as the latter is a special case (a subset) of the former. Hence, if the circumstances allow for (n+1) plausible plaintexts, then the *variable size no limit* key system will always be able to find a key that will match each and every one of these (n+1) plausible messages with a proper key. In that respect the variable size no limit system is as effective as the message size key system. But this theoretical equivalence is of little interest.

The variable size key system will deserve attention only if it helps relieve the tedium and burden of the message size key.

In other words, one would attempt to use a moderate size key, K_0, to encrypt message P_0 into C, and subsequently be able to point to another plausible (decoy) message P_1 which will correspond to same C via another key (decoy) K_1. And same for additional plausible messages P_2, P_3, ...P_n.

Since the size of the key is part of the system secrecy, the user will always be able to match P_1, P_2, ...P_n with corresponding keys which are as long as C.

So, theoretically this will work. Only that if the size of K_0 is much smaller than C (which is the essential attraction of the system), and K_1, K_2, K_3, ...K_n are all of the length of C, then it would be difficult to argue that any of P_1, P_2, P_3, ...P_n are the real message. P_0 will stand out.

This brings us to the notion of the deniability key list.

The Deniability Key List

Given a cryptographic system based on a variable size and no preset limit key, and given a ciphertext C generated by such system, one may wish to list all the plausible plaintext messages P_0, P_1, P_2, P_3, ...P_n and their smallest corresponding keys: K_0, K_1, K_2, K_3, ...K_n in the order of key size.

Let P_0 represent the actual message that was encrypted into C (the intended reader and the writer both use K_0). Let the rest of the indices: 1,2,3,...n be assigned by sorted size order. So that for i=1,2,...(n-1), it will hold that the size of K_i is not larger than the size of $K_{(i+1)}$:

K_1 -- P_1
K_2 -- P_2
K_3 -- P_3
K_4 -- P_4
K_5 -- P_5
.....
.....
.....

Let S(K) be the size of K, then: if $S(K_0) << S(K_1)$, the formal deniability of the system is not very effective in practice. Say, if $S(K_0) << S(C)$ and $S(K_i) = S(C)$ for i=1,2,...n, then K_0 stands out as the

right key. We must recall that for all plausible messages P_i there exists at least one key K_i of size S(C) that would match Pi with C.

Say then that the above ordered key list determines the practicality of the system.

If there exists a value j > 1 such that:

$$S(K_j) \leq S(K_0) \leq S(K_{j+1})$$

Then the system offers good deniability. A cryptanalyst who will uncover the list (but will be aware only of C, and not of K_0) will have to conclude that messages $P_1, P_2, P_3, ...P_j$. are at least as plausible as P_0 (the real message), since they employ keys of equal or smaller size. This will amount to an effective deniability of order j.

Realistically if $S(K_{j+1}), S(K_{j+2}), ..S(K_{j+t}) \rightarrow S(K_0)$ then the effective deniability order will be j+t. All this assumes that the messages $P_0, P_1, P_2, P_3, ...P_n$ are of equal likelihood (or above a threshold likelihood) as far as any non-cryptographic evidence might suggest. In such cases the size of the key is the only discriminator.

The Key-List Condition

All this leads us to conclude that for a variable size key cryptography to claim a measure of Shannon Security, it must employ a key which is not much smaller than the smallest key that will fit for another plausible plaintext. We designate this condition as the key list condition.

Implementation Procedures

Having derived the key-list condition we may now point to the message size (or ciphertext size) as the critical factor in devising a practical procedure for the variable size key.

Naturally if C is of small size then the size of K_n (the largest key in the key list) is also small (K_n is at most as large as C), and hence all the keys in the key list are of plausible size.

It is for large C that there is an issue to consider. Generally the larger C, the more difficult it should be to find a practically small key that would serve as decoy, or deniability argument.

Cryptographic procedures call for a-priori key identification (and distribution). That means that K_0 must be selected before P_0 is known. It must be selected so that the resultant C will allow for one to build the key list so that K_0-P_0 will fit somewhere inside it. This need, for any variable size system will determine the aggregate size of C that a given key K_0 should generate.

A good variable size key system is one where a single small size K_0 will generate even a large C such that the user (or anyone else) will be able to construct a sufficiently large key list and fit K_0 inside.

In practice one may look for a stream cipher with a “closed loop” or say, expansion-reduction

(reversal) algorithm. (See below).

Existence and Discovery

The key list condition poses a question of existence and discovery. Given a system of variable size key, does it satisfy the key list condition for a given cipher and set of plausible messages? And if it does, how difficult is it to discover that list? A subsequent question is: who should discover it?

As a minimal requirement a user might argue that existence is sufficient, since the adversarial cryptanalyst will pick it up as a matter of course. What is necessary, is that the opponent will come up with the list and be confused by it.

A more rigorous user will want to build the list on his own. This requirement creates a unique situation. The user and his opponent will be busy with the same task: building the key list. The user will know (P_0,K_0), but will have to crack his own ciphertext to identify (P_1,K_1), (P_2,K_2), etc. Which is exactly what the opponent will have to do. Only that the opponent will need to also find the (P_0,K_0) pair, and then discriminate between it, and the decoys.

This duality brings into question the desirability of easy discovery. It is not clear whether the user would wish that effort to be small or large. The former option will help the user build the key list; the latter will make it difficult for the opponent to do the same.

In general, the larger K_0, the more likely it is to find the key list, since there exist a better chance to find a decoy key of equal or smaller size. Thus to insure deniability of high order, it is advisable to use large K_0.

A variable-size key cryptography might pose the question: *is a given key list -- the best?* Is K_i the smallest key that would link plaintext P_i with ciphertext C? Unlike the prevailing cryptographies, the key variability in the variable-size case may lead to a situation where there are several keys that would link a (P_i, C) pair. A good list will put forward the smallest ones. For a given cryptography one might concern himself with the question of proving that a given key is indeed the smallest.

For every proposed variable-size key ciphersystem one would be advised to analyze the existence and the discovery effort of an effective key list.

Expansion-reduction Stream Cipher Algorithms

Normally a stream cipher is constructed as a one way function whereby a small seed generates an expanded binary stream which is used as a pseudo-random message-size key. Elsewhere in cryptography one is searching for the reverse: one-way reduction algorithms (Hash functions). For the purpose at hand the interest is focused on a two-way algorithm which can be used for both purposes: expansion and reversal reduction. Such an expansion-reduction algorithm will serve as a

regular stream cipher system, but will also facilitate a search for small seeds that would serve as decoy keys to achieve what we may call: ***Meta Shannon Security***; namely: Shannon Security of a comparatively small order, which nonetheless offers a fundamental advantage: it can not be brute-force exhausted. Its security can not be torn down neither by raw computer power, nor through clever new mathematics.

Conclusion

This paper analyzed the existence and desirability of cryptographic systems based on variable size key of no preset limit. It has shown that the much coveted Shannon security can be extended to such systems, and while the extent of such security is not as extensive as is offered by the One Time Pad, it is nonetheless substantial, and it offers a novel advantage: achieving Shannon Security (alternatively called deniability) without having to employ the unwieldy large keys which are necessary for OTP. The combination of mathematical security and small or moderate size keys renders these cryptographies into a class of great promise and much interest.

Further Reading

Meta Shannon Security algorithms are featured in Daniel and related products distributed by D&G Sciences – Virginia Technology Corporation, P.O.Box 1022 McLean VA 22101-1022 http://www.dgsciences.com