Pattern Devoid Cryptography

Making "Brute Force" the sole attack strategy -- And Defending Against it

Hidden Math threatens cyber security; Randomness is the answer

Gideon Samid
Electrical, Computer and System Engineering
Computer and Data Sciences
Case Western Reserve University, Cleveland, OH
Gideon.Samid@CASE.edu

Abstract: Pattern loaded ciphers are at risk of being compromised by exploiting deeper patterns discovered first by the attacker. This reality offers a built-in advantage to prime cryptanalysis institutions. On the flip side, risk of hidden math and faster computing undermines confidence in the prevailing cipher products. To avoid this risk one would resort to building security on the premise of lavish quantities of randomness. Gilbert S. Vernam did it in 1917. Using modern technology, the same idea of randomness-based security can be implemented without the inconvenience associated with the old Vernam cipher. These are Trans Vernam Ciphers that project security through a pattern-devoid cipher. Having no pattern to lean on, there is no pattern to crack. The attacker faces (i) a properly randomized shared cryptographic key combined with (ii) unilateral randomness, originated ad-hoc by the transmitter without pre-coordination with the recipient. The unlimited unilateral randomness together with the shared key randomness is set to project as much security as desired up to and including Vernam levels. Assorted Trans Vernam ciphers (TVC) are categorized and reviewed, presenting a cogent message in favor of a cryptographic pathway where transmitted secrets are credibly secured against attackers with faster computers and better mathematicians. A vision emerges: a cryptographic level playing field, consistent with the emerging culture of Web 3.0.

Table of Contents

1.0 Introduction

The group think of modern cryptography is that cipher builders are better mathematicians than cipher crackers, hence if the former don't see a mathematical cryptanalytic pathway, neither would the latter. The fact that Alan Turing proved them wrong eighty years ago makes no difference, that is the power of group think [58]. Following the revelations of Edward Snowden though, more people suspect that the cryptographic powerhouses are using unpublished math to compromise security of their targets [59]. A new line of thoughts emerges: building ciphers that are not based on pattern-loaded algorithms, but rather on the opposite: pattern-devoid algorithms. [25, 31,32, 35, 40, 47, 49, 53, 56, 57].

The search for pattern is never exhaustive; hidden layers loom, and are fertile grounds for attackers aiming at mathematics-reliant ciphers. Randomness -- on the other hand, is by definition the absence of pattern. Pattern may be viewed as the holes, folds, and protrusions on a mountain you climb, you hang on to them on your way up. Randomness is a perfectly smooth wall, there is nothing to hook up to.

Cryptographers though, are mathematicians in heart, pattern is their thing. They loathe and dismiss the plain logic that argues: pattern-reliant ciphers are threatened by deeper pattern missed by the cipher builder and spotted by the cipher attacker.

The more pattern you detect, the more pattern must be suspected -- fertile ground for attackers. It is hard for a defendant to credibly appraise the mathematical insight of his attacker. It is much more feasible to estimate adversarial computing power. This leads to a cipher construction strategy wherein mathematical prowess will be dethroned as the main cryptanalytic tool, and only computing power -- brute force -- will be left. Such ciphers don't necessarily have to be as perfectly secure as the illustrious Vernam cipher. Their risk of compromise though, must be credibly appraised in terms of the required computing workload. Such appraisal, together with a credible estimate of the attacker's computing power, will generate a good overall estimate of security over time.

We therefore are off on our way to search for ciphers that will compel their attacker to resort to brute force cryptanalysis, finding mathematical superiority useless. We focus first on symmetric ciphers.

Mathematical cryptanalysis works its way backwards. The ciphertext, C, is examined and studied, to reveal its generating plaintext, P, and its operational key, K: C = E(P,K). The strength of a cipher depends on the mathematical properties of the encryption algorithm E, and on the randomness load of the key, K. The stronger E is, the smaller (the weaker) K can be. (E.g.: ECC keys are smaller than RSA keys because ECC is regarded mathematically more robust than RSA).

We are on a hunt for a cipher that will not rely on the mathematical properties of E for its security (because those properties are vulnerable to hidden mathematical insight), but rather rely on the randomness facing the attacker.

We don't have to look far, 104 years ago, Gilbert S. Vernam patented his now famous "Vernam Cipher" [34] where security is 100% based on the randomness of the key. The Vernam cipher is not based on any hackable mathematical properties. Its security relies on the purity of the randomness of the key. And as is well known, Vernam security is perfect, as has been proven some 25 years later by Claude Shannon [54].

The price paid for this high security was a key as large as the plaintext. This represented too much inconvenience at the time, and hence cryptographers steered away, and took the technology of secrets in the opposite direction: using small keys combined with mathematical complexity. This is where we stand today. This trend has developed so much momentum that sidekicks suggesting an alternative, are all but ignored.

Initial application of the emerging Artificial Intelligence Assisted Innovation (AIAI) system [5, 60] show how fertile it is to reinspect innovative forks of the road, and revisit the path not taken. Which is exactly what was done with respect to the old Vernam road junction.

Let R be the randomness used by a cipher to achieve its aim. Let M be the mathematical complexity used by a cipher to achieve its aim (clearly a nebulous definition). Our premise is that M is inherently not a reliable secret-protective-source because the greater the mathematical complexity of M, the greater the chance for a lurking mathematical breach strategy, which will be spotted by the attacker, not by the builder of the cipher. R, on the other hand, is 100% reliable, to the extent that R is perfectly random. The greater R, the greater the security. If $|R| \geq |P|$, P- the encrypted plaintext, then the cipher may admit perfect security.

Gilbert S. Vernam set $|R|=|K|$, [34]. Vernam's randomness was captured in the shared key between the transmitter and the recipient of a message. But this is not a necessary requirement. The transmitter may use a-priori unshared, unilateral randomness, U, to achieve:

$$|\boldsymbol{R}| \leq |\boldsymbol{K}| + |\boldsymbol{U}|$$

What is needed for this idea to fly are two things:

1. The attacker should not be able to distinguish between K and U

2. The Recipient should not be confused by the impact of U.

If these two conditions (1,2) are fulfilled then the communicators will be able to achieve any desired security, with a limited size key, K, together with sufficiently large U.

If we set:

$$|\boldsymbol{R}| = |\boldsymbol{P}| - |\boldsymbol{K}| = |\boldsymbol{C}| - |\boldsymbol{K}|$$

We achieve Vernam security in apparent violation of Claude Shannon's dictate [10]: |K|=|P|.

Shannon's proof of mathematical secrecy was based on comparing an attacker holding the ciphertext, C, to an attacker that only knows the size of the ciphertext [54]. If the two attackers face the same challenge then there is no advantage to having knowledge of C over having only knowledge of the size of C -- which is what Shannon defined as perfect security.

If the C-knowing attacker has to check out |K| options, and the *C-not-knowing* attacker has to check |P| options then if |K|=|P| there is really no advantage to knowing C. However if |K| < |P|, then the C-knowing attacker has an advantage over the C *not-knowing attacker*.

Three observations: (i) If |K| < |P| but still very large, then security is not 'perfect' but may still remain formidable. One could seek to construct ciphers wherein the security deterioration as the used plaintext exceeds the protective randomness, is happening very slowly; (ii) what if |K| is not known to the attacker? Or say, if |K| is unbound? In that case the attacker cannot conclusively end the search of the key space.

The third observation is the crux of this treatise. Vernam is a cipher where security receives no contribution from mathematical complexity, M=0; its security is generated only by the randomness of the key. We can therefore define a class of ciphers, to be called "Trans Vernam" which have this very property: their security is not generated from mathematical complexity but solely from the amount of randomness used. The more randomness -- the better security, for a key larger or equal to the encrypted plaintext, this security is perfect.

An attacker of a Trans Vernam cipher is cornered to use Brute Force attack only, and hence the cryptanalytic burden ahead is represented by the quantity of randomness, R, facing the attacker. Vernam generated the required R through the shared key, K, but this is not a requirement. R can be generated from a shared key K and unshared randomness U. U will be randomness generated by the transmitter without pre coordinating with the recipient.

The attacker of the Trans Vernam Cipher facing R, (of unknown size) will have no information as to how R divides to K and U. Perhaps U is zero, and R=K? Therefore the brute force attacker will have to check every R option, counting $2^{|R|}$ possibilities (again, |R| is not known to the attacker). The security of the ciphertext is determined first by the transmitter and the recipient together, setting up the key, K, and then by the unilateral determination of the transmitter as to the value of U. The transmitter indeed is the best party to decide on how much security a ciphertext deserves, and hence how much U to use, since the transmitter knows how sensitive the transmitted message is. The transmitter determines the value of R, and hence controls the security projected from the unleashed ciphertext.

It is worth emphasizing that the secrecy regarding the size of the key is critical for the Trans Vernam Strategy to work. If the key size is known, and everything else is known to the attacker (Kerckhoffs' principle, [61]) except the contents of R, then the attacker could train their brute force on K and void the impact of U.

The larger the size of the key, K, relative to U, the greater the chance that a smaller key will offer a plausible (and false) decryption of the ciphertext. And hence, an attacker, finding a reasonable key, is inherently plagued by the doubt of it being a false key, steering the attacker to a misleading decryption of the ciphertext.

For this scheme to work, it is necessary for the recipient to be able to use their knowledge of K to decrypt C to P without being confused by the impact of U. There are several established ciphers that accomplish this. These are Trans Vernam Ciphers.

Sources: [41, 46, 47, 57].

2.0 Characterization of Trans-Vernam Ciphers

Two shared characteristics: (i) the size of the key is part of its secret, (ii) the Trans Vernam key is reusable.

A third differentiating characteristics: (iii) ciphertext absorbing, or not absorbing the unshared randomness.

With the size of the key being part of its secret, the brute force attacker will never be able to conclusively terminate their search, even if a good key candidate was found. It may be a mistake, and the right key is a larger one. This uncertainty can be used by the communicators by sending meaningless random bits between them. Their attackers will exhaust themselves looking for ever higher keys to crack the communication.

The Vernam key is not re-usable. New key bits must be furnished to encrypt new plaintext. This generates a daunting synchronization challenge which is a basic reason for the unattractiveness of Vernam. However Claude Shannon's proof does not require non-reusability, it only requires key size. Trans Vernam Ciphers operate with a large enough key, which is being used over and over again.

To wit: Using Vernam, if the overall plaintext P is divided to smaller sections P_1, P_2, ... P_t, then the respective Vernam key K, will have to be divided to same size keys: K_1, K_2, ... ,..... K_t, and encryption proceeds as:

$$C_i = E_{Vernam}(P_i, K_i) \text{ ..... } for\ i=1,2,...t$$

In a Trans Vernam cipher one proceeds as follows:

$$C_i = E_{Trans\text{-}Vernam}\ (P_i, K) \text{.... } for\ i=1,2\ldots t$$

So no synchronization is needed, and more importantly a Trans Vernam cipher may conveniently be used by multiplicity of communicators without requiring all parties to follow on all communications among other parties, as the case is with Vernam.

Some Trans Vernam Ciphers (TVC) generate the required randomness through a sufficiently large key, and some use unshared randomness, U, with small more manageable keys. The use of U may be reflected in a ciphertext larger than its generating plaintext $|P| < |C|$. The cipher is designed such that the recipient can readily ignore the inflated part of the ciphertext, and credibly extract P from C, using K.

When a size-preserving cipher, E, relies on a key K which is smaller than the message P, then it means that out of $2^{|P|}$ possible plaintexts of size |P| bits only $2^{|K|}$ are viable. The rest are not. The reduction from $2^{|P|}$ possibilities to $2^{|K|}$ possibilities is effected by the pattern inherent in E. This pattern is in the cross hair of the non-brute force attacker. To the extent that $|K| \rightarrow |P|$ that is the extent that pattern shrinks, and security shifts to randomness. When |P| spills over |K| the security of the cipher deteriorates. However the rate of deterioration can be credibly appraised by the cipher users, who will decide at each instant if the risk-benefit balance will make it worthwhile to continue using the same key, or arrange for a replacement. It is a main objective for the Trans Vernam Cipher designer, to construct a cipher where said deterioration is as slow as possible,[19, 25].

Ahead we discuss the two main categories of TVC: (i) ciphertext inflated TVC, and (ii) ciphertext-not-inflated TVC, followed by means to handle the extra burden of an inflated ciphertext.

3.0 Uninflated Ciphertext TVC

Identifying three ciphers of the ciphertext uninflated category. One is based on transposition, another on a multi-dimensional roadmap taking the role of the shared key. The third is based on a scheme by which the communicators will iteratively replace an existing key with a new key without this replacement being compromised by the attacker.

3.1 Complete Transposition Cipher

The following is a short overview of the cipher, defined in "Equivoe-T: Transposition Equivocation Cryptography." [18], "The Ultimate Transposition Cipher (UTC)." [43], and in "Equivoe-T: Transposition Equivocation Cryptography" US Patent 10,608,814 [62].

A plaintext P of size n=|P| bits will have $T = n! / (n_0! * n_1!)$ permutations, where n_0 and n_1 are the number of 0 and 1 in P respectively: $n_o + n_1 = n$. The highest value of T is $T_{max} = n! / 2 (0.5n)!$. By using a key such that:

$$K_{max} \geq T_{max}$$

one achieves complete permutation equivocation.

It is easy to beef this security up to full Vernam by constructing a string Q as:

$$Q = P \oplus \text{"11....1"}$$

and concatenating: $\pi = Q \| P$

π has 2n bits, n of them are 0 and n are one. It has T = (2n)! / 2n! permutations, hence a key space where:

$$K_{max} > (2n!)/2n!$$

will elevate the security of P to Vernam grade. But unlike Vernam the transposition key K_T does not need synchronization and it can be used for plaintexts smaller than itself. What is more, if |P| grows larger than |K| then security level drops down from perfection, but this drop down happens very slowly so that high security is maintained.

The complete transposition cipher uses a transposition algorithm with a distinct advantage. Most transposition algorithms are size defined. Namely a simple mapping list will dictate how n bits will be reshuffled around to different places, but these reshuffling instructions specify a particular value for n. The complete transposition cipher is defined over any value of n. It defines a sequence to shipping bits from P to another list, P^t of same size where the order by which the bits are picked for shipping is determined by the value of the key. The algorithm is symmetric and P^t will be readily returned into P.

Clearly, like with Vernam, the complete transposition cipher has M=0. Its security is not based on any mathematical complexity. It is based solely on the fact the transposition key K^t is randomly selected from a large enough key space.

It has been proven that the space for a single integer key, K, operating through the complete transposition algorithm will cover all the permutations. As described above if P is separated to parts P_1, P_2, ... P_t then the same transposition key, K will transpose each P_i to P_i^T over its bit size $|P_i|$ without need to synchronize.

3.2 Space Flip Cipher

This is a brief overview of the cipher described in detail in "SpaceFlip: Unbound Geometry Cryptography" [36, 63], "SpaceFlip: Unbound Geometry Security" US Patent 10,790,977 [37]

This cipher fashions a key in the form of dimensionality-undetermined space, S, comprising a distance geometry [2]. Namely each of the points of the space has a random distance to any other point. This space, is clearly not a metric space, and does not obey proximity laws. The points comprising the space are letters of an alphabet. A line on this space is defined as a sequence of points where the next point in the sequence is constrained by the points that make up the line so far. The determination of the next point is dependent on all the distances from this point to all the points not already on the line. Every point has a next point until all the points are part of the line. Thereby each line can be regarded as a permutation of the points in S.

To send a plaintext letter A, the transmitter may randomly choose a letter B, and mark a line from it. This line will encounter letter A after s steps. Therefore the combination {B,s} will be interpreted as the letter A, by the recipient who is working with the same space S. Next time when A is to be sent out as a plaintext letter the transmitter may choose another letter, say, D, and mark a line off it. This line will encounter the letter A after s' steps, so the combination {D, s'} will be interpreted as A by the recipient aware of S.

By choosing the alphabet large enough, the security will be robust enough -- again only through randomness, no mathematical complexity. The 0.5t(t-1) distances marked on S comprising t points are randomly chosen, and so is each ciphertext letter. By choosing as alphabet all the possible p bits long strings (an alphabet comprising $2^p = t$ letters) the communicators determine the size of S and the level of the projected security.

To be accurate the ciphertext for this cipher is roughly twice as large as the plaintext, but this should be considered a moderate increase. This cipher, SpaceFlip, can also be implemented with size preservation, only with less security. The transmitter will randomly determine a step count value, s, and then communicate every plaintext letter A with the letter A' such that A appears as the s point on the line in S that begins with A'. This can be run t times without repetition, making it for that measure equivalent to Vernam, without the inconvenience of Vernam.

We will discuss in the next section how SpaceFlip can be implemented with a size increase option.

3.3 Forever Key Cryptography

This is a brief overview of the cipher described in detail in "FAMILY KEY CRYPTOGRAPHY: Interchangeable Symmetric Keys; a Different Cryptographic Paradigm" [20], "SpaceFlip Plus: Ordinal Cryptography" US Patent 11,159,317 [66]

This cipher is constructed so as to allow for infinite number of keys to be interchangeable, namely have the same effect. Each of these so called 'family of keys' K_1, K_2, ... K_i, will encrypt a given plaintext to the same given ciphertext. On its face it seems counterproductive: an attacker will now have an infinite number of keys to hunt, any one of those keys will compromise the cipher. Why make it easier on the attacker?

The answer is plain. An attacker that would successfully compromise a transmission and will extract the plaintext from the ciphertext is likely to have spotted some key K_a which is different from key K_u, which the users have used. The most that an attacker can accomplish is to figure out the entire family of interchangeable keys. There is no way for the attacker to nail down which of the infinite number of keys was actually used by the users. The ciphertext simply does not contain any information that points to the particular key that was used to generate it. That is the power of interchangeable keys; they conceal the identity of the key that was actually used.

This advantage that the users hold over their attacker can be used to exchange a transformation formula, to compute a derived key K'_u from the used K_u, and continue their secret communication with their new-shared key, K'_u. The key derivation formula is constructed such that every input will generate a different output. And since the attacker does know the identity of K_u, they would not be aware of the identity of K'_u either, although the transformation formula is exchanged in the open.

The users can then continue their communication using the derived key K'_u', while dismissing K_u. From the point of view of the attacker, this will be as if the users agreed on a new key in secret. The attacker will not be able to exploit any information garnered from cryptanalyzing K_u to learn anything about K'_u.

After some use the communicators will repeat this operation and derive a third key, K"$_u$, and so on. Even if the attacker cracks the communication that use some key K*$_u$, the identity of K*$_u$ is not extracted, and hence when K*$_u$ is being transformed to K*'$_u$, the identity of the new key is not known either, so the users operate as if they started to communicate with their original key, K$_u$. In other words, this is a mechanism to keep a finite key for indefinite use.

In practice the number of keys to be considered by the attacker is not infinite because there is a practical limit to how large such a key can be. But this implies that the users can approach this infinite protection by choosing keys that are larger. Since the only way to crack this cipher is by using brute force, the users can credibly estimate how much plaintext they can safely use before their family of keys will be flashed out by the attacker. Based on this estimate the users will switch to the next key before that measure of plaintext is processed.

Because in practice the keys in the family of keys are of a finite count, then the security of this 'forever key' cipher is not infinite either. Albeit, its level of deterioration can be credibly appraised. And when needed the transformation of the keys will build very large keys, so that security is upheld.

4.0 Inflated Ciphertext TVC

In this category ciphers pack the unilateral randomness injected by the transmitter into a containing ciphertext that hence is getting bigger. This is the price paid for security that is not vulnerable to hidden math. As long as the recipient can readily shake off the extra ciphertext material, the only inconvenience with these ciphers is the much larger file to be communicated as ciphertext (no need to store the inflated ciphertext). With today's technology this is not a big burden even for large documents. When it comes to images, audio and video files such a large size multiplier for the ciphertext does present a problem. We will see ahead how to navigate this hurdle.

We review here the following ciphers: (i) SpaceFlip, (ii) BitMap, (iii) BitFlip, (iv) The Unary Cipher.

4.1 Increased Ciphertext SpaceFlip

This is a brief overview of the cipher described in detail in "SpaceFlip: Unbound Geometry Cryptography" [36, 63], "SpaceFlip: Unbound Geometry Security" US Patent 10,790,977 [37]

SpaceFlip as described above (in the "3.0 Uninflated Ciphertext" section) can be deployed in a ciphertext-increased mode. The procedure is as follows. To transmit a letter A to the recipient the transmitter will randomly choose any letter from the alphabet A', and then randomly choose an integer g from an arbitrary g-space from 1 to g_{max}. Next the transmitter will randomly choose (g-1) integers in the range $1 \leq r_i \leq t$, for i= 1,2,...(g-1) where t is the number of points in S.

Next the transmitter will mark a line L_1 from A' to the letter r_1 steps ahead, say, letter A". From letter A" the transmitter will mark a line comprised of r_2 steps, ending up with letter A"', from there to the next letter r_3 steps away. Repeating the same sequence for all the (g-1) distances, the transmitter will end up at some letter B. The line from B will encounter letter A r_g steps ahead. The transmitter will now send over to the recipient the letter A' and the g distance integers: r_1, r_2, ... r_g.

Marking this information on the shared space S, the recipient will spot the letter A very readily.

There is no limit to the value of g. Large g values lead to large ciphertexts. As the key is being used more and more, the transmitter uses more and more unshared randomness, picking larger and larger g values.

Here too, no pattern, the distances between the t points on S are fully randomized. There is no math to crack, brute force is the only viable attack strategy, and hence the size of S is the sole source of security. The users can stop using a given space S (a key) when the amount of plaintext used through it is exceeding a security threshold. This SpaceFlip protocol will achieve Vernam security with the convenience of a Trans-Vernam cipher.

4.2 BitMap

This is a brief overview of the cipher described in detail in "At-Will Intractability Up to Plaintext Equivocation Achieved via a Cryptographic Key Made As Small, or As Large As Desired - Without Computational Penalty." [6], "BitMap Lattice: A Cyber Tool Comprised of Geometric Construction", US Patent 10,911,215, [8]; "Denial Cryptography Based on Graph Theory." Gideon Samid (2004) US Patent 6,823,068. [13]

This cipher is a map where the points are associated with letters of a particular alphabet. Points are connected to their direct neighboring points but not to other points. The connections themselves may be viewed as "walking bridges" allowing a traveler to walk from one point to the other. Every bridge is marked by a letter of the same or of a different alphabet that marks the points on the map. The basic idea of the cipher is that the plaintext is viewed as a travel guide. Each successive letter in the plaintext indicates the next travel destination. A plaintext comprising p letters will then be associated with a travel path on this map, comprising p visited spots. And since the passage from spot to spot requires passing through a bridge, and each bridge is marked with a letter, then the same pathway that was identified by the plaintext, is similarly identified by listing the successive bridges traversed by the traveler. It is a simple principle: a path on a map can be described by a list of successive destinations, or equivalently by a list of successive bridges one walks on. The plaintext is seen as a travel guide pointing to the visited destinations; the ciphertext, by contrast, is seen as the list of bridges one passes through when taking the same path. Each bridge is associated with a letter, so the pathway described as crossed bridges will manifest itself as a series of letters -- the ciphertext.

The cipher is designed so that any possible plaintext can be mapped into a pathway, and every possible pathway can equally be described by a list of crossed bridges. The transmitter will mark the points (spots) on the pathway corresponding to the plaintext, then describe the same pathway by marking the crossed bridges, and when the list of crossed bridges is assembled, it is communicated to the recipient.

The recipient on his part will use the ciphertext to mark the same pathway on their copy of the map. Once marked the recipient will read out the visited spots and mark the letters represented by these spots in a sequence -- thereby reconstructing the plaintext.

The attacker without possession of the map (the key) will not be able to reverse the ciphertext into the plaintext. The configuration of the map is randomized; the marking of the points on the map and the markings on the bridges are all highly randomized, subscribing only to a weak restriction. Thereby the map projects no analytic complexity. It is fair to say that only brute force has a prayer and a hope to crack this cipher.

It is worth noting that the attacker does not know how big the map is. Each spot and each bridge can be visited and crossed countless times. So a very small map will encrypt and decrypt a very long message if necessary without betraying the size of its key.

Building a large key is providing share randomness, K. Albeit, BitMap will allow a transmitter to inject unshared randomness as follows: the plaintext alphabet A' is deemed to be comprised of l -1 letters. Another letter, letter number l is then added by injecting it between any two successive letters that are identical (in the plaintext). This operation removes all instances where two instances of same letter are written one after the other. The transmitter can now replace any letter in the l letters alphabet, A, of the plaintext, with any number of identical letters next to each other. This inflates the plaintext to any desired size. Doing so will in turn lead to an inflated pathway and an inflated ciphertext. The recipient though, recovering the long plaintext, will simply shrink all letter repetition to a single letter and thereby extract the original plaintext.

Illustration: the string ABC will become AAAABBBBBBCCCCCC. This inflated plaintext will then mark a much longer pathway on the map. This longer pathway will be translated to a long ciphertext that would confound the attacker. The intended recipient will extract the inflated ciphertext but will shrink it to size. Every string of consecutive same letter, like AAAA and BBBBBB, will be replaced by a single same letter: AAAABBBBBBCCCCCC → ABC.

The attacker, in possession of the ciphertext does not see the duplication. It does not show on the bridge-list. As a result the attacker wrestles with a long ciphertext, not knowing which parts, if any, are the real concealed message and which parts are confounding noise.

4.3 BitFlip

This is a brief overview of the cipher described in detail in "BitFlip: A Randomness Rich Cipher" [7]; "BitFlip Cyber Demonstration" [65]; "Transmitter for Encoding Information with Randomly Flipped Bits and Transmitting That Information Through a Communication Channel" US Patent 10,728,028, [76]; "Advanced BitFlip: Threat Adjusted, Quantum Ready, Battery Friendly, Application Rich Cipher" US Patent 10,541,808, [77].

The idea behind BitFlip is to have a large number of ciphertext letters map into a single plaintext letter in parallel to having a large number of ciphertext letters map into no plaintext letter. The first attribute resists pattern recognition through randomized selection of ciphertext letters among the many which map into a given plaintext letter. The latter attribute allows the user to freely inflate the ciphertext with false letters, which the intended reader will readily recognize as such, while the attacker will have to regard them as message bearing. By carefully subjecting all choices to randomization, the users expunge any pattern from the construction of the cipher, cornering the attacker to brute force attack strategy -- the efficiency of which can be credibly assessed by the BitFlip users.

BitFlip works on some alphabet A comprising l letters. Each letter, L_i (i=1,2,..l) is represented by a bit string of size n bits -- selected randomly. Each letter is associated with "Hamming distance" value, h_i, where $0 \leq h_i \leq n$. A ciphertext letter c is a bit string of size n. Letter c is decrypted to plaintext letter L_i iff:

$$H(c, L_i) = h_i$$

where $H(c, L_i)$ is the Hamming distance between c and L_i.

Ciphertext letters that don't decrypt to any of the l plaintext letters are discarded.

The selection of the n-bit string representation for each of the l letters is randomized. The selection of the l h_i values is randomized, the selection of the ciphertext letter that decrypts to a given plaintext letter is randomized. The peppering of the ciphertext with so called decoy ciphertext letters that are meaningless, is also randomized. There is no thread of pattern to crack here. The attacker is cornered to brute force attack strategy.

4.4 The Unary Cipher

This is a brief overview of the cipher described in detail in "A Unary Cipher with Advantages over the Vernam Cipher" [3]; "Unary Cryptography Demonstration Site" [64]; "Mixed Unary Cryptography" US Patent Application 17/323,908.

A bit string x indicating an integer of value v, can be expressed through a string of (v+1) "0", concatenated to list of (r+1) "1", where r is the number of leading zeros in x. This, so called unary expression is larger in bit count, but it is otherwise equivalent to the original expression. The two can be derived one from the other.

This property can be used as follows: a plaintext P is randomly broken down to n consecutive substrings: P_1, P_2, ... P_n. The value of n, and the size of the substrings $|P_i|$ for i=1,2,...n is randomly selected, as long as:

$$|P| = \Sigma\ |P_i| ...... \text{ for } i= 1,2,...n$$

Each P_i is then mapped to its corresponding unary expression, thereby writing P in a combined unary fashion, P*. P* may be peppered with "0,1" element because these elements vanish when transposed back from unary format.

P* is then transformed with a complete transposition key as discussed above: $P^* \rightarrow P^{*T}$.

P^{*T} is the ciphertext. $P^{*T} = C$

It can be shown that any value of P' from some high level Q > P to zero may be encrypted to the same ciphertext C, thereby projecting functional equivalent with Vernam. Its advantage over Vernam is that the key (the transposition key) can be used over and over again, no synchronization needed.

P* may be wrapped with a header of the form 00....1 and a trailer of the form 11.....0. These are two more options to pad the ciphertext in a way that would not confuse the intended reader, but will build a growing cryptanalytic burden.

To further increase the cryptanalytic burden one will prepare the pre-transposition string as one with equal count of zeros and ones, as follows: (i) compute $P\text{^} = P \oplus 11......1$ (ii) concatenate P with P^: $P^{**} = P^* \,||\, P\text{^}$, then transpose P** (of size 2|P|).

To the extent that the transposition operation is complete, this cipher projects Vernam grade security over its secret size key.

5.0 Split Security Solutions

The price paid for randomness-based security is a large key, and in many cases a large ciphertext. In order to keep projecting high security from the same key, k, the ciphertext will have to be longer and longer than the plaintext. This increased ciphertext length is looming to become a more and more serious problem for large plaintexts. When the materials to be encrypted are audio files, images, or video files then the much larger ciphertext may be prohibitive. This challenge can be handled via Entropic Impact Discrimination.

5.1 Entropic Impact Discrimination

This is a brief overview of the cipher described in detail in "Split Security Solutions", US Patent Application 17/510,324 [78]

A plaintext P may be divided to meaning-bearing elements m_1, m_2, ... m_t. Each meaning-bearing element is associated with an entropic impact that reflects the advantage gained by an adversary in case this element is exposed. The elements may then be divided to low impact elements which have an entropic impact below a given threshold and high-impact elements which have an entropic impact above that threshold. The latter are slated to be encrypted with a Trans Vernam cipher, and the former are encrypted with a size preserving cipher. Thereby the users decide how much inconvenience (large ciphertext) to put up with, in order to get a given level of security.

The high impact selection may be automatic. For example, facial recognition software will identify faces in an image or in a video, and mark these faces for TVC encryption. The rest will be encrypted with size-preserving ciphers. In the worst case scenario the adversary will see what the image shows, but will not figure out who the people in the image are.

6.0 Spontaneous Cryptography

In the 1970s cryptographic science has made a dramatic leap ahead. It enabled two strangers to practice cryptographic protocols despite having no prior shared key to rely on. Spontaneous cryptography, or cryptography between strangers, revolutionized the practice of commerce; it became a key enabler of our migration towards cyber space. Alas, the prevailing schemes are heavily reliant on mathematical pattern, which very likely hides deeper patterns with which to crack this cryptography. So in order to maintain the great benefits introduced by spontaneous cryptography it is suggestive to investigate constructing bilateral secrecy on randomness.

Here too, a scroll back to the pages of history proves helpful. Much as Vernam revisited helps us with ordinary encryption, so Ralph Merkle is a new pointer for spontaneous encryption. Unlike his followers Diffie and Hellman, Ralph Merkle based his original idea for strangers developing a bilateral secret while exposed on the network, not on mathematical complexity but rather on a temporary advantage claimed by two communicators, who can solve a riddle a bit faster than their attacker. The communicators then use this temporary secret to secure a permanent secret. Merkle's idea was for one communicator to present the other a list of difficult computational tasks, for which the submitter already knew the answers. The recipient chooses randomly one of the various computational tasks, computes it, and communicates the result to the sender. The value of the answer tells the transmitter which one of the tasks the recipient chose, and that information qualifies as a temporary secret until the attacker will compute the entire list and also find out which task the recipient chose.

An advanced version of Merkle's randomness (not pattern) based spontaneous cryptography is offered by FigLeaf.

6.1 FigLeaf

This is a brief overview of the cipher described in detail in "Randomized Bilateral Trust (RABIT): Trust Building Connectivity for Cyber Space (FigLeaf)" U. S. Patent 10,798,065, [79].

The FigLeaf idea is based on the familiar "birthday paradox": it turns out that a group of only 23 people have a 50% chance to include two people with the same birthday, month and day of the month. Similarly two strangers would agree to randomly pick n mathematical items from a large list L of such items. The value of n can be adjusted to make it x% likely for the two items selectors to have picked the same item (a picked item remains in L). The two then start a dialogue. The selected items have various properties. One communicating stranger randomly selects a property and tells the other the n values of this property in the n items she selected. The other communicator can then exclude any of his selections for which the value of this property is not on the list submitted by the first communicator. His list shrinks n → n'. Next the list-recipient, selects another property, and hands over the list of its n' values. The first communicator will delete from his list all the items that have a value for that property that is not in the submitted list. This will shrink her list n → n". By repeating this protocol the two strangers would either realize that they have no item in common, and in that case they will restart the protocol, or they would both identify the one item they both randomly picked. It will take an outside observer much longer time to use the information exposed in the protocol to spot the shared item. Any of the unused properties of the shared item will qualify as a temporary secret, which the communicators will use to secure a permanent secret key if necessary. For cases like money transfer the temporary secret will do. Once the money is transferred the shared secret becomes useless.

Unlike the Diffie-Hellman scheme, FigLeaf is randomness based, and the only route of attack is brute force.

7.0 Extended Applications

The power of cryptography to hide a secret in an exposed capsule (a ciphertext) serves a variety of applications beyond enabling a conversation in a hostile environment. Most common among them are (i) cryptographic authentication, applicable to humans and things alike, (ii) pattern concealment, and (iii) graded randomness applications. To the extent that mathematical cryptography is vulnerable to the original application, it is similarly vulnerable to its extended applications. And to the extent that Trans Vernam Ciphers cure the vulnerability of nominal cryptography for secret communication, it similarly cures the vulnerability presented itself when used to authenticate a document, a person, a thing.

Presenting (i) authentication applications, (ii) pattern concealment.

7.1 Authentication

Authentication is a process where a Verifier verifies that another party, "The Prover", is in possession of a piece of information, P. The challenge is repetition: to ensure that the proof does not disclose to an attacker how to falsely claim bona fide possession of P. Hence, direct exposure of P is not an option. There are three prevailing ways to hide P: (i) pass P within a secure channel, (ii) apply private-public key, (iii) apply randomness. The second option is by far the most popular. Yet, it is the first target for quantum cryptanalysis, and its days are numbered. Good alternatives are in order.

Presenting: (i) challenge-response authentication (ii) randomized authentication protection

7.1.1 Challenge-Response Authentication

The following is a short overview of the cipher, defined in "Efficient Proof of Knowledge of Arbitrarily Large Data Which Remains Undisclosed" US Patent 10,594,480 [67]

To prove possession of a body of data, P, by a "Prover", the "Verifier" will randomly pick a number R, and a function f, and pass both to the prover (in the open). The prover will use R, and f to transform P into a bit string Q: Q = f(R,P). Next the prover will parcel out Q to n distinct concatenated substrings q_1, q_2, ... q_n.

$$Q = q_1 \,||\, q_2 \,||\, ... \,||q_n$$

The breakdown of Q to n substrings will be carried out according to a shared rule which will ensure that:

$$q_i \neq q_j, ... \text{for } i{\neq}j, \text{ and } i,j = 1,2,....n,$$

The breakdown rule is not secret. The prover will then apply the Complete Transposition Cipher, as described above, to reshuffle the n substrings to a transposed Q: Q^T.

Q^T will be passed to the verifier. The verifier will similarly identify the n substrings q_1, q_2,.... q_n, and confirm that Q^T can be constructed from q_1, q_2,.... q_n in some order.

One may note that the verifier may verify the prover being in possession of P, without the verifier being in possession of P. All that the Verifier needs in order to do their job is to have n substrings: q_1, q_2,.... q_n, without having knowledge of the particular permutation thereto that assembles them to Q.

Since the transposition is complete, the attacker in possession Q^T will have first to list all the possible ways, m, in which Q^T can be divided to an unknown number of substrings, n_1, n_2, ... n_m, compliant with the substrings division rule, then for each possible number of strings consider all the permutations thereto.

By selecting f, and R so as to generate Q of any desired size, the verifier will ensure that the brute force load on the attacker will exceed their ability to extract P in a timely manner. Because the combination of R and f is not repeated, the attacker will have to extract P from Q^T in order to prove possession thereof.

7.1.2 Protecting Authentication Databases

The following is a short overview of the ciphers, defined in "Method for Inhibiting Mass Credentials Theft" US Patent 10,395,053 [70]

Identity authenticators happen to be organizations serving a large number of customers. Using secure channels these authenticators receive the credentials of their customers, compare them to their records, and thereby authenticate them. The records kept by these organizations are at the cross hair of sophisticated attackers, since a single penetration will net private data of all the customers of the victim organization. This can be prevented by allowing the authenticators to perform the authentication without keeping their customers' data in their authentication records. It seems impossible at first glance, how would one authenticate unknown data? It can be done through zero knowledge techniques that are math-loaded. Here we present a randomness-based solution.

Information submitted for authentication is written as bit strings. Bit strings are conveniently written in Base64. Base64 is a language comprised of 64 letters. We map letter number i in this alphabet (i = 1,2,...64) to a bit string comprising i bits -- disregarding the identity of the bits. We now determine the identities of these message bearing bits through a source of randomness. This will yield a fully randomized layout, R, of the message A submitted for authentication. Let's divide A to n bits segments: $A = a_1 || a_2 || ..... a_m$ where $|a_i| = n$. Each segment will also be randomized. Let us now flip $h < n$ bits in each segment. Such flipping will not affect the prime message, written in Base64, but it will shift the bit expression of the Base64 letters. Each message a_i will be shifted to a'_i, where a_i and a'_i exhibit a Hamming distance h between them:

$$h = Hamming(a_i, a'_i)..... \text{ for } i=1,2,...,m$$

We now can store the shifted expression: $A' = a'_1 || a'_2 || ..... a'_m$ in the server's database, and send the customer the message $A = a_1 || a_2 || ..... a_m$ to use when authenticating herself.

The prime message (written in Base64) will be the same both for the record kept on the customer's phone and for the record kept in the server's database. When the server receives the message from the customer, the server first authenticates the prime message (account number, name, password etc.), and then examines the bit identities of the submitted bit string.

If the Hamming distance between the customer’s record and the server’s record for all sections of A is h, then the transaction goes through. If the test fails, the transaction stops, and a response protocol is activated.

Should a hacker break in to the server and copy its records, they will not harvest the customers' record. They will get a hold of the server's data. Should a hacker attempt to steal a customer's identity using her credentials that were stolen from the server, then the server will immediately realize that the Hamming distance between its records and the data submitted for authentication is zero and not h. An alarm will sound to alert the server and conclude that a hacker pretends to be a customer. The server is further alerted to the fact that the server was compromised. Once so realized the server will simply refresh its records, maintain the prime message but change the Hamming distance from h to $h' \neq h$. This simple act will void the hacker's harvest. The stolen data which has a Hamming distance h from the customer's record will not enable the hacker to contrive credentials that would exhibit a Hamming distance h' from the servers' records. That is because the hacker does not know which bits were flipped and which were not -- this choice was made randomly. The net effect of this tool is that (i) a breach is instantly discovered, and (ii) is readily recovered from. This recovery is swift and painless without bothering the customer.

7.1.3 Authentication of Material Items

The following is a short overview of the technology defined in "Proving Material Identity with Quantum Randomness -- Financial and General Applications" US Patent 10,754,326. [74], "BitMint Hard Wallet: Digital Payment without Network Communication: No Internet, yet Sustained Payment Regimen between Randomness-Verifiable Hard Wallets" [75].

Counterfeiting material items is an advanced fraud industry, affecting mainly manufactured items of value. Governments are in a race with counterfeiters over banknotes, passports, and various licenses and documents. Manufacturers suffer when the market is flooded with look-alike products that steal their customers and destroy their reputation.

Much as the prevailing cryptography is opting for greater and greater mathematical complexity to fend off cryptanalysis, so do manufacturers, adding hologram signatures and other physical complexities to remain one step ahead of the counterfeiters. We have seen how randomness is an alternative solution strategy against cryptanalysis; the very same principle applies to material authentication: chucking the unified complexity race in favor of randomized complexity.

A pattern-loaded manufacturing complexity will eventually be deciphered. And once so the counterfeiter will flood the market with hard to detect counterfeits. Applying randomness, each item has its own unpredictable signature, so there is no one 'secret' that works for all items of the same kind. The counterfeiter will have to tailor the counterfeit to individual signatures. What is more, the randomized signature is comprising a very large number of measured properties. Some are published on a public ledger, allowing the verifier to compare the published and the measured. Albeit, many more properties are not published a-priori, and only released to the public upon demand. The verifier will check the submitted item against the just released properties -- a counterfeit will fail the test.

7.2 Pattern Concealment

The following is a short overview of the cipher, defined in "Effective Concealment of Communication Pattern (BitGrey, BitLoop)" US Patent 10,673,822, [71]

In many practical circumstances attackers gain a consequential advantage by analyzing the pattern of communication traffic: who writes to whom, when, how much, how often, etc. While encryption per se hides the content of the communication, it does expose its pattern. Trans-Vernam ciphers can be used to cure this deficiency. TVC may inflate the ciphertext at will, while the intended reader will not be confused by the meaningless bits and properly interpret the meaningful bits, as described herein. This situation may be exploited by establishing a fixed bit rate among the communicators. The bit flow will range from no-messaging, all bits are meaningless, to all-messaging, no bits are meaningless, and any state in between. The communicators will read only the messages intended for them, but attackers will see a steady unchanging bit flow rate, remaining in the dark as to whether anybody talks to anybody, or who talks to whom, how often and how much.

8.0 Randomness Technology

The most popular source for randomness are algorithms that generate bits sequences that comply with a given (arbitrary) set of rules. John Von Neumann said that anyone generating randomness from algorithms does not understand, neither randomness, nor algorithms. Indeed it makes little sense to abolish mathematical complexity with randomness that is itself a product of mathematical complexity. There are two classes of non-algorithmic randomness: (i) physical complexity, (ii) quantum randomness. The former is based on the formidable amount of real time knowledge that must be processed in order to defeat it, and the latter is based on a first principle of quantum mechanics.

It is noteworthy that perfect randomness can be theorized, not proven. No matter how many tests are conducted over a source of randomness, the results can always be explained as coming from a source where the deviation from perfect randomness is too small to detect in this finite test. This fact casts a thin but present shadow on the assertions for security claimed herein.

Presenting: (i) quantum randomness technology, (ii) physical complexity randomness technology.

8.1 Quantum Randomness Technology

The following is a short overview of the technology, defined in "Rock of Randomness" US Patent 10,467,522, [73] and in "The Rock of Randomness: A physical oracle for securing data off the digital grid" [42]

Commercial outfits today offer elaborate apparatuses generating quantum grade randomness [72]. What is needed then is (i) robust packaging, and (ii) effective duplication, (iii) copy protection. These three needs have been satisfied through "The Rock of Randomness". It packs randomness in the chemical structure of the material constituents of the rock. The data, hence, is off the digital grid, which means it is beyond the territory that is subject to digital compromise. One needs to have access to the Rock itself, in order to read its data. The Rock packs very large quantities of data in its molecular composition. It is measured in an analog format, then digitized. Even a small piece of rock may pack an enormous amount of data, beyond what is practical to image in a database.

The Rock is not vulnerable to accidental physical punishment, nor to happenstance chemical obstruction. Melting will destroy it, but otherwise it keeps. The manufacturing of the Rock can be duplicated, but given a manufactured Rock it is infeasible to duplicate it. Communicators holding a duplicate of the same Rock each, will enjoy the full power of Trans Vernam Ciphers.

8.2 Physical Complexity Technology

The following is a short overview of the technology, specified in US Patent Application #17063523.

Quantum randomness enjoys the credibility of the most elevated scientists who claim it to be perfect. Only that the generators of this randomness are embedded in complex electronics, which is subject for attack. So while the created bit flow is unbiased, the bit sequence that is poured to its consumers may be contaminated. This is one argument in favor of a closer source based on sufficient real-world complexity that is per symmetry tests devoid of any pattern [29]. One such source is a contraption wherein insulating bubbles rise in a conductive liquid, and reduce its effective conductivity. The reduced conductivity suppresses the bubbles flow, (feedback cycle) which in turn increases the effective conductivity, that now increases the flow of bubbles. The mechanism varies the quantities of the rising gas, as well as its distribution over a range of bubble size. The conductivity variance is translated into a randomized bit stream.

This bubbles randomizer will be external to the consuming computers, and be readily replaceable. Unlike a quantum source, this complexity apparatus hinges on a feedback cycle, so that any disturbance will be diffused to high quality randomness.

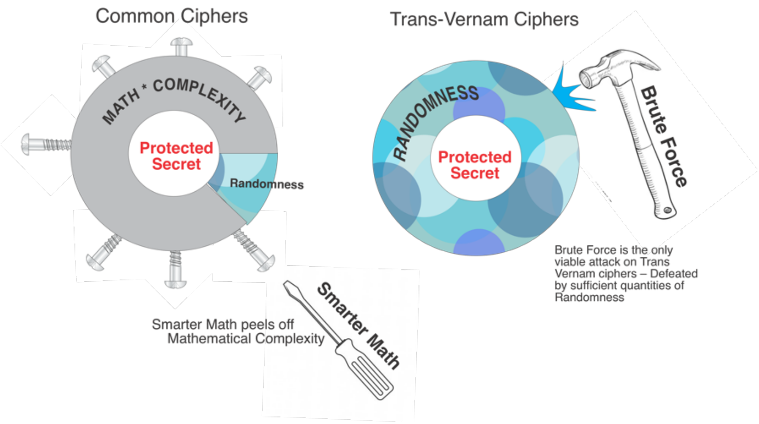
* * *

A Note on the Innovation Solution Protocol: This presentation is a case study for the innovation path known as historic retrace in which one traces the innovative history of a present state, revisits innovative forks of the roads, and examines roads not taken. The idea being that in hindsight the unselected options in that junction point may look more attractive than when first encountered. Pursuing those untried avenues is a choice loaded with possibilities. Progress from natural evolution to science and technology is a zig-zagging path. This thesis emerges from a rigorous practice of the Innovation Solution Protocol (Innovation^{SP}) [5, 60]. It identified the 104 years old Vernam cipher as the fork in the road from where cryptography selected the small-key path, which was the wise choice for the technology of the day. Albeit with the tools we have at present, the road not taken -- projecting security through randomness, not through math -- is the road that leads to new cyber vista.

Allegory: The following short tale illustrates the message contained in this article. Two pals, Alice and Bob play a game of dice. One throws, the other guesses. A right guess will move 1$ from the dice thrower to the dice guesser. Playing for hours both players end up with just about the same amount of money they brought to the match. This is very disappointing for Alice who is much better educated than Bob, and knows everything there is to know about probabilities and computing. So she proposes to Bob to introduce a tiny variation to the game. Instead of throwing one dice, they will each throw two. Instead of guessing in the range 1-6, they will be guessing in the range 2-12. Bob innocently accepts, but no sooner do they switch to two dice than Alice cleans Bob wallet to his last dollar. While innocent Bob randomly guesses a choice from 2-12, Alice uses her probability education and guesses 7 every time. Bob is losing money every night, blaming his bad luck.

One bright day Bob realizes that his losses occurred when the game was switched from one dice to two dice, so he insists on returning to the original mode. Lo and behold Alice advantage vanishes.

Trans Vernam ciphers -- return to Vernam philosophy (not to the Vernam protocol) -- are tantamount to innocent Bob returning to the one dice mode where no matter how smart, or how stupid a player is their playing field is level.


Common Ciphers
MATH * COMPLEXITY
Protected Secret
Randomness
Smarter Math peels off Mathematical Complexity
Smarter Math
Trans-Vernam Ciphers
RANDOMNESS
Protected Secret
Brute Force
Brute Force is the only viable attack on Trans Vernam ciphers – Defeated by sufficient quantities of Randomness

Outlook

For decades national security agencies around the world operated on the premise that they are one step ahead of their adversaries in terms of mathematical insight, and computational power. They have opted for a state of affairs where their adversaries trust and use a cipher that these agencies can secretly compromise on account of their technological edge. If the targets of these agencies will shift from pattern-loaded to pattern-void encryption, then this decades-old paradigm will exhaust its efficacy. From the public point of view, Trans Vernam Ciphers do level the playing field. Ordinary citizens will be able to kick-start spontaneous cryptography and then communicate with mathematically guaranteed security, withstanding any assaults by math and machine; randomness is unassailable.

This prospective empowerment of the global individual citizen comes about just when the vision of Web 3.0 is picking up steam. Perfect timing.

Reference

1. “An extension of the Shannon theory approach to cryptography”. Martin Hellman, IEEE Transactions on Information Theory, V. 23, 3 1977, pp. 289 - 294
2. "A New Perspective of Geometry and Space as an Evolutionary Organizer of Data." Gideon Samid, http://www.dgsciences.com/Geometry_H7n18.pdf
3. "A Unary Cipher with Advantages over the Vernam Cipher" Gideon Samid, https://eprint.iacr.org/2020/389
4. "Anonymity Management: A Blue Print For Newfound Privacy" Gideon Samid, The Second International Workshop on Information Security Applications (WISA 2001), Seoul, Korea, September 13-14, 2001 (Best Paper Award).
5. "Artificial Intelligence Assisted Innovation" Gideon Samid, https://www.intechopen.com/online-first/artificial-intelligence-assisted-innovation
6. "At-Will Intractability Up to Plaintext Equivocation Achieved via a Cryptographic Key Made As Small, or As Large As Desired - Without Computational Penalty." Gideon Samid, 2002 International Workshop on CRYPTOLOGY AND NETWORK SECURITY, San Francisco, California, USA September 26 – 28, 2002.
7. "BitFlip: A Randomness Rich Cipher" 2017, Gideon Samid, Sergei Popov, https://eprint.iacr.org/2017/366.pdf
8. "BitMap Lattice: A Cyber Tool Comprised of Geometric Construction", US Patent 10,911,215, Feb 2, 2021
9. "Chaos-based Cryptography: A Brief Look Into An Alternate Approach to Data Security" A Sharif, NI Raihana, A Samsudin - Journal of Physics 2020 https://iopscience.iop.org/article/10.1088/1742-6596/1566/1/012110/meta
10. "Communication Theory of Secrecy Systems". Claude Shannon (1949) http://netlab.cs.ucla.edu/wiki/files/shannon1949.pdf

11. "Cryptography of Things: Cryptography Designed for Low Power, Low Maintenance Nodes in the Internet of Things" Gideon Samid https://search.proquest.com/openview/8897dc1c4858b327796917b8fcdff7ae/1?pq-origsite=gscholar&cbl=1976348
12. "Cyber Passport: Preventing Massive Identity Theft. " Gideon Samid, https://eprint.iacr.org/2016/474
13. "Denial Cryptography Based on Graph Theory." Gideon Samid (2004) US Patent 6,823,068.
14. "Drone Target Cryptography" Gideon Samid, https://eprint.iacr.org/2016/499
15. "Effective Concealment of Communication Pattern (BitGrey, Bitloop)" US Patent 10,673,822 June 2, 2020
16. "Encryption Sticks (Randomats)" Gideon Samid ICICS 2001 Third International Conference on Information and Communications Security Xian, China 13-16 November, 2001
17. "Encryption-On-Demand: Practical and Theoretical Considerations" Gideon Samid https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.215.2463&rep=rep1&type=pdf
18. "Equivoe-T: Transposition Equivocation Cryptography." Gideon Samid, International Association of Cryptology Research, ePrint Archive https://eprint.iacr.org/2015/510
19. "Essential Shannon Security with Keys Smaller than the Encrypted Message the Encrypted Message" Gideon Samid, https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.75.1585&rep=rep1&type=pdf
20. "FAMILY KEY CRYPTOGRAPHY: Interchangeable Symmetric Keys; a Different Cryptographic Paradigm" Gideon Samid https://eprint.iacr.org/2021/458
21. "Feeding Cryptographic Protocols with Rich and Reliable Supply of Quantum-Grade Randomness" Gideon Samid, https://eprint.iacr.org/2020/968
22. "Feeding Cryptographic Protocols with Rich and Reliable Supply of Quantum-Grade Randomness" Gideon Samid, https://eprint.iacr.org/2020/968.pdf
23. "Fingerprinting Data" Gideon Samid, https://eprint.iacr.org/2018/503
24. "Hush Functions Extended to Any Size Input versus Any Size Output." Gideon Samid, https://eprint.iacr.org/2012/457.pdf
25. "Intractability Erosion: The Everpresent Threat for Secure Communication" Gideon Samid The 7th World Multi-Conference on Systemics, Cybernetics and Informatics (SCI 2003), July 2003.
26. "Larger Keys, Less Complexity" Gideon Samid, A Strategic Proposition." https://eprint.iacr.org/2018/406.pdf
27. "Proposing a Master One-Way Function." Gideon Samid, https://eprint.iacr.org/2007/412
28. "Randomized Bilateral Trust (RABIT): Building Connectivity for Cyber Space" US Patent 10,798,065, Oct 6, 2020
29. "Randomness as Absence of Symmetry" Gideon Samid, THE 17TH INTERNATIONAL CONFERENCE ON INFORMATION & KNOWLEDGE ENGINEERING (IKE'18: JULY 30 - AUGUST 2, 2018, LAS VEGAS, USA) http://bitmint.com/SymRand_Vegas_H8518R.pdf
30. "Randomness in digital cryptography: A survey" K Marton, A Suciu, I Ignat - Romanian journal of information science 2010 https://www.academia.edu/download/46676431/Randomness_in_Digital_Cryptography_A_Sur20160621-25262-h5ar54.pdf
31. "Randomness Rising - The Decisive Resource in the Emerging Cyber Reality" Gideon Samid, Int'l Conf. Foundations of Computer Science | FCS'18 | https://www.bitmint.com/RandomnessRising_GSamid_H1o16.pdf
32. "Re-dividing Complexity between Algorithms and Keys" Gideon Samid, International Conference on Cryptology in India, 2001 - Springer https://link.springer.com/chapter/10.1007/3-540-45311-3_31
33. "Rivest Chaffing and Winnowing Cryptography Elevated into a Full-Fledged Cryptographic Strategy" Gideon Samid, 2018, Proceedings of the International Conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government (EEE); Athens, (2018). https://search.proquest.com/openview/8ea94f941732d85fb24512d5e7582820/1?pq-origsite=gscholar&cbl=1976356
34. "Secret Signaling System". US Patent 1310719A. Gilbert S. Vernam (1918)
35. "Shannon Revisited: Considering a More Tractable Expression to Measure and Manage Intractability, Uncertainty, Risk, Ignorance, and Entropy" Gideon Samid, https://arxiv.org/abs/1006.1055
36. "SpaceFlip: Unbound Geometry Cryptography." Gideon Samid, https://eprint.iacr.org/2019/285.pdf

37. "Spaceflip: Unbound Geometry Security" US Patent 10,790,977, Sept. 29, 2020
38. "T-Proof" Gideon Samid https://img.chainnews.com/paper/71f69315d015d9fc5dd4ffbc97f87aab.pdf
39. "T-Proof: Secure Communication via Non-Algorithmic Randomization." Gideon Samid, https://eprint.iacr.org/2016/474
40. "Tailored Key Encryption (TaKE)" Gideon Samid, https://eprint.iacr.org/2000/011.pdf
41. "The Myth of Invincible Encryption" Gideon Samid, Digital Transactions May-June 2005
42. "The Rock of Randomness: A physical oracle for securing data off the digital grid": Gideon Samid, Gary Wnek, Material Research Society Bulletin 09 April 2019
43. "The Ultimate Transposition Cipher (UTC)." Gideon Samid, https://eprint.iacr.org/2015/1033.pdf
44. "Threat Adjusting Security" Gideon Samid, https://eprint.iacr.org/2018/084.pdf
45. "Transmitter for Encoding Information with Randomly Flipped Bits and Transmitting That Information Through a Communication Channel", US Patent 10,728,028 Jul 28, 2020
46. "User Centric Cryptography" Gideon Samid, Proceedings of the International Conference on Security and Management (SAM); Athens, (2018) https://www.proquest.com/openview/a60ecf397b6c46373356a1d4369dce5d/1?pq-origsite=gscholar&cbl=1976342
47. "What a 100-year-old Idea can teach us about Cybersecurity" World Economic Forum, Nov 2017 https://www.weforum.org/agenda/2017/11/what-a-100-year-old-idea-can-teach-us-about-cybersecurity
48. "When Encryption is Not Enough--Effective Concealment of Communication Pattern, even Existence (BitGrey, BitLoop)" Gideon Samid, https://eprint.iacr.org/2019/556
49. “Algorithmic Randomness and Complexity” School of Mathematics and Computing Sciences, Downey, R, Hirschfeld, D. Victoria Univ. Wellington, New Zealand. http://www-2.dc.uba.ar/materias/azar/bibliografia/Downey2010AlgorithmicRandomnes s.pdf
50. “Communication Theory of Secrecy Systems” Claude Shannon http://netlab.cs.ucla.edu/wiki/files/shannon1949.pdf
51. “Computability and randomness” Niels A. The University of Auckland, Clarendon, Oxford, UK, 2008
52. “Deniable Encryption” Rein Canetti, Cynthia Dwork, Moni Naor, Rafail Ostrovsky CRYPTO '97Volume 1294 of the series Lecture Notes in Computer Science pp 90-104Date: 17 May 2006
53. “Probabilistic Encryption” Goldwasser, Micali, Jr. of Computer and System Science, Vol 28, No 2, pages 270-299
54. “Shannon’s Proof of Vernam Unbreakability” https://www.youtube.com/watch?v=cVsLW1WddVI
55. “STRENGTHENING THE SECURITY FOUNDATION OF CRYPTOGRAPHY WITH WHITEWOOD’S QUANTUM-POWERED ENTROPY ENGINE” Richard Hughes, Jane Nordhold http://www.whitewoodencryption.com/wp-content/uploads/2016/02/Strengthening_the_Security_Foundation.pdf
56. “Survey on Cryptographic Obfuscation” Ma´ t´ e Horva´ th 9 Oct 2015 International Association of Cryptology Research, ePrint Archive https://eprint.iacr.org/2015/412
57. “The Unending Cyber War” Gideon Samid, DGS Vitco ISBN 0-9635220-4-3 https://www.amazon.com/Unending-Cyberwar-Gideon- Samid/dp/0963522043
58. “The Code Breakers” David Kahn, The MacMillan Co. 1967.
59. “Edward Snowden: The Untold Story” Wired Mag. Aug 14, 2014
60. “The Innovation Solution Protocol” (Innovation^{SP}), https://InnovationSP.net
61. “Kerckhoffs' Principle“ http://www.crypto-it.net/eng/theory/kerckhoffs.html
62. “Equivoe-T: Transposition Equivocation Cryptography” US Patent 10,608,814 March 31, 2020.
63. “SpaceFlip: Unbound Geometry Cryptography” Gideon Samid https://dblp.org/rec/journals/iacr/Samid19.html
64. “Unary Cryptography Demonstration Site” https://UnaryCryptography.com
65. “BitFlip Cyber Demonstration” http://wesecure.net/learn/BitFlipEncrypt.php
66. “SpaceFlip Plus: Ordinal Cryptography” US Patent 11,159,317 * Oct 26, 2021
67. "Efficient Proof of Knowledge of Arbitrarily Large Data Which Remains Undisclosed" US Patent 10,594,480, March 17, 2020.
68. "Cyber Companion: Attaching a Secondary Message to a Primary One" US Patent 10,541,954 Jan21, 2020
69. "Live Documentation (LiDO)" US Patent 10,733,374, Aug 4, 2020

70. "Method for Inhibiting Mass Credentials Theft" US Patent 10,395,053 Aug 27, 2019
71. "Effective Concealment of Communication Pattern (BitGrey, BitLoop)" US Patent 10,673,822, June 2, 2020
72. “Quantum Random Number Generation” https://www.idquantique.com/random-number-generation/overview/
73. "Rock of Randomness" US Patent 10,467,522 Nov 5, 2019
74. "Proving Material Identity with Quantum Randomness -- Financial and General Applications" US Patent 10,754,326. Aug 25, 2020
75. “BitMint Hard Wallet: Digital Payment without Network Communication: No Internet, yet Sustained Payment Regimen between Randomness-Verifiable Hard Wallets” Gideon Samid, 2020 IOT, Electronics and Mechatronics Conference (IEMTRONICS), IEEE International.
76. “Transmitter for Encoding Information with Randomly Flipped Bits and Transmitting That Information Through a Communication Channel” US Patent 10,728,028, July 28, 2020.
77. “Advanced BitFlip: Threat Adjusted, Quantum Ready, Battery Friendly, Application Rich Cipher” US Patent 10,541,808, January 21, 2020.
78. "Split Security Solutions", US Patent Application 17/510,324, Oct 25, 2021
79. "Randomized Bilateral Trust (RABIT): Trust Building Connectivity for Cyber Space (FigLeaf)" U. S. Patent 10,798,065, October 6, 2020.