"Tesla Cryptography:"

Powering Up Security with Other Than Mathematical Complexity

Tesla Pivoted the Automobile Industry From Oil To Batteries.
Can Something Similar Happen To Cryptography?

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

Abstract: For decades now, mathematical complexity is being regarded as the sole means to creating a sufficient distance between a ciphertext and its generating plaintext. Alas, mathematical complexity operates under the irremovable shadow of stealth cryptanalysis. By its nature mathematical complexity is vulnerable to smarter mathematicians and better equipped adversaries, which is a sufficient motivation to explore an alternative means to project security. Applying the Innovation Solution Protocol such an alternative has been found: randomness. Not as next to mathematical complexity, rather as its replacement. Unlike complexity, randomness is not vulnerable to smarter mathematicians and better equipped adversaries. It removes the shadow under which all modern ciphers operate by proposing a framework wherein the message transmitter may apply arbitrary quantities of ad-hoc randomness with which to secure a transmission over a secret key of arbitrary large size; and where only a part thereto may participate in any instance of encryption; and where security is increased in proportion to the amount of randomness involved. Handling the large quantities of randomness is 'messy' and inconvenient, albeit, the user, not the cipher designer, decides how much inconvenience to put up with in order to build sufficient security to meet the pressing threat. With sufficient randomness, transmission security may exceed One-Time-Pad (OTP) in as much as even the size of the plaintext is not determinable. Ciphers that shift the security responsibility to the user are called "Trans-Vernam", honoring Gilbert S. Vernam's OTP, or "Tesla Ciphers", reflective of the fact

that Tesla offered a new power source for the automotive industry, much as the Tesla ciphers offer a new security source for cyberspace. The Tesla cryptographic modality has its security substantiated with a mathematical proof. It is Quantum ready and AI resistant. It is battery-friendly, and ultra fast. Albeit this proposal brings to question a long-established cryptographic premise, with all that is involved.

1.0 Introduction

It is a given, unchallenged, taken as self evident: the wall that prevents one from extracting the plaintext from the ciphertext is made up of mathematical bricks. The more bricks the better. Cryptography hinges on mathematical complexity. The cryptographer builds the wall, the cryptanalyst breaks the wall, a cat and mouse game that repeats itself through the course of history.

Yet, there is one human legacy that is older and more profound than the mathematical foundation of cryptography: the imperative of innovation. Innovation begins with a question. Profound innovation begins with a question challenging a foundational concept. We may note that modern cryptography is shaped up by the innovation that followed a simple question: *is it necessary for the encryption key and the decryption key to be one and the same?*

Let's see then how far we can go following a question challenging the premise that mathematical complexity is the unopposed means to keep the plaintext at a distance from the corresponding ciphertext.

What else is there? Is the next question. But before exploring it let's pose a practical question: what is wrong with mathematical complexity, it served us well. *If it ain't broke -- don't fix it.*

We address this suitability issue first. Come to think about it, mathematical complexity by its essence is vulnerable to a smarter mathematician that will detect a hidden pattern, an invisible simplicity that would collapse the complexity assumed by the cryptographer. This would be devastating. And it is ever present. AES-256 is the most popular cipher in the world. Security consultants point to the fact that no one published a breach thereto. If so many smart cryptographers try for years and fail -- this is solid evidence in favor of trusting the cipher.

We are reluctant to think about it, but we have nothing better to advocate for AES-256 than the fact that no one claimed to have cracked it, and many sure tried. This argument is faulty on its face. Since the whole world is using this particular cipher and trusting it with their top secrets then the profit from cracking it will be incalculable -- many times more than the accolades one would expect from publishing the breach. If any agency has cracked this popular cipher, then the political power they serve will have an enormous strategic advantage on the world scene today. It is therefore obvious that the NSA and its many counterparts are putting their brightest people and their most powerful computing machines for the singular purpose of breaking AES. They would be liable for negligence of duty otherwise. Have they succeeded?

The people who know the answer to this question are not talking, but one thing is for sure, whether they have already defeated the cipher, or expect to do it in the foreseeable future -- their public position will be that AES is unbreakable, and all suggestions to the contrary are pipe dreams. Remember, Churchill sacrificed British lives for the sole purpose of preventing the Germans from suspecting that their cipher, Enigma, was broken. Having spent a fortune in effort, talent and dollars to break AES - if the world ever suspects that it happened, the world would abandon it and make the expensive breach useless.

It brings us to the conclusion that the fact that no one published a breach and the fact that a majority of cryptographers and cryptographic agencies claim it is unbreakable is no evidence for the veracity of these claims.

There is one counter argument. Apart from the secret cryptographic agencies that remain silent, we do enjoy a vibrant academic and corporate community of cryptographers, which are beholden to no political organization, and are committed to academic freedom and bona fide security. These top-of-the-line cryptographers honestly try to crack the cipher (as well as other popular ciphers) and their failure, one claims, is to be taken as solid evidence in favor of the strength of the cipher.

Indeed so, however the mindset and the methodologies used by the prime cryptanalysis shops are far different from the efforts taken by a single bright professor, even if helped by a cadre of PhD students. I successfully discouraged a student that asked me to be a thesis professor for his attempt to find a breach to a certain cipher. "What if you work on this breach for five years, and fail -- on what ground will you claim your PhD?"

Agency work is based on smart allocation of efforts in many fronts. It hinges on a subtle connection between pure mathematical analysis and computational power. It is not committed to an elegant comprehensive breach formula, rather it is a chase for 'weak keys' that admit a breach. Increasingly human reasoning is augmented with AI analysis -- further empowering the breach seekers. Enormous resources are brought to bear, way beyond the academic attempts.

Cryptography today is ruled by the better mathematicians and the better equipped cryptanalysts. A hierarchy of power is formed with the NSA widely regarded as topping the pyramid. A prevailing argument claims that the National Security Agency is satisfied with this order, as it tries to maintain its technical advantage over adversarial governments and unfriendly organizations around the world.

For the vast majority of users of cryptography this hierarchy puts them in a fishbowl. Powerful organizations, friendly and unfriendly, may be reading their mail and thereby violating their privacy, all the while being protected behind a veil of deniability.

It is a stark conclusion: data assets protected by mathematical complexity are permanently under the shadow of stealth breach by one or many smarter and better equipped cryptanalysts. This is especially troubling since we migrated to cyberspace, and the people we trust and wish to converse privately with are geographically apart, so we must rely on cryptography to establish basic privacy, and cryptography of any color cannot disengage from the shadow of being compromised by people we don't trust.

And come to think about it, the average user is stuck with a cipher the application he uses chose to deploy. In the best case, the user can select between two or more ciphers, but whatever cipher they choose, they are stuck with the security they project. They can't tweak it.

So realizing, we now turn to the first question raised before: *what else is there?*

What can match the power of mathematical complexity to keep a distance between a ciphertext and its generating plaintext?

We need not go far. Inspecting the computing machines that are about to overshadow Turing machines, we find that their distinction is rooted in randomness, which we may note also expresses our understanding of physics and nature itself. Randomness and probability rise as a more effective computational paradigm than determinism and repeatability. We exchange 0 for negligible probability, and replace 100% with probability close enough to 100%. Since cryptography is being attacked by randomness and probability rooted machines, why not let it defend itself with the same elements?

The idea to extract security from randomness may be enticing, but one would tend to take a smaller step, and look for innovative ways to improve security, the baseline of which is founded on mathematical complexity. In other words, one would seek to do *more* with randomness, not *instead* of mathematical complexity, but rather to complement it.

Such a smaller step may look prudent, alas, as long as mathematical complexity remains essential for the projected security, the respective cipher will suffer from the very same 'shadow

threat' of stealth cryptanalysis. The only way to escape this lingering shadow is to remove mathematical complexity from the essential ingredients of the projected security. This reasoning challenges one to find a way to handle randomness so innovatively that it can do the job without reliance on complexity at all. Is it asking too much?

As a guide to answering this question let's consider the following situation:

Alice and Bob play a guessing game with two dice, one throws them, the other guesses the outcome between 2 and 12. Alice happens to be a bright mathematician, but Bob is mathematically challenged. So while Bob consults his stomach and guesses in turn all the numbers from 2 to 12, Alice always guesses 7, which is 6 times more likely than, say 12, or 2. Each game comprises 100 rounds and Alice wins every game -- by a lot! One day Bob makes a small change. Instead of tossing two dice, they each toss one dice at a time, now guessing between 1 and 6. Lo and behold, from that moment on Alice lost the consistent advantage she had over Bob. She is still much smarter than him, but the new game voids her smarts advantage. The random nature of the dice forces a level playing field.

Now, this is not exactly a typical cryptographic challenge, but it points to the idea that proper use of randomness may indeed allow everyone, however dumb, to communicate securely despite being attacked by much smarter and much better equipped adversaries.

Before going for the ambitious undertaking to search for a way to replace complexity with randomness, let’s imagine the way cyber security will operate under these conditions.

We identify two classes of randomness: *pre-shared* and *unilateral*. Alice and Bob share a randomly selected key, K. The transmitter of a message may use ad-hoc randomness, A, in the transmission procedure. This unilateral randomness is well handled by the recipient, but presents a cryptanalytic barrier before the attacker. In a randomness-based security climate, the amount of randomness, K+A, determines the security projected by the generated ciphertext. Alice and Bob may pre-share a series of keys K_1, K_2. ... of successively larger sizes, allowing the transmitter to

indicate to the recipient which key to use for every transmission based on its sensitivity and the appraised threat. Additionally the transmitter may respond to a sudden rise in the prevailing threat, and kick in larger and larger amounts of ad-hoc randomness, to ensure the security of their data assets despite the rising threat.

This scenario highlights a crucial distinction in favor of randomness-based security. While a math-based cipher user cannot tweak the algorithm to improve the security, the randomness-based user can deploy as much unilateral randomness that is needed in their estimation.

The above scenario described a fundamental change in cryptographic roles. Power is shifted from the distant designer of a mathematically complex cipher to the actual user, operating the cipher. The user is in the field, they are most aware of the prevailing threat. The user knows the sensitivity of the data they encrypt. The user is the one to be harmed if their data integrity is violated or exposed. The user is the prime stakeholder, it serves them well to be in control of the security that their data projects. This role-switch on its own creates a powerful motivation for us to find a way to deploy randomness as a replacement for mathematical complexity.

2.0 Projecting Security in Proportion to The Amount of Deployed Randomness

We find three categories of randomness in a typical cipher setup: the key, the ciphertext, the encryption protocol. The key is a true random selection from a key space, the ciphertext is 'fake randomness': it looks random but it carries pattern - a message, the plaintext, and the procedural randomness is dictated by the encryption algorithm, which may include random inputs (e.g. "nonce").

Our aim is to change the way we regard these three categories of randomness in order to use them to project security without resorting to mathematical complexity.

2.1 The Key

Every cipher has a randomized key. Alas, the keys that we commonly use are (i) small, and (ii) of well-defined size, which is firmly linked to the encryption and decryption algorithm. This suggests that every cipher we use is subject to brute force cryptanalysis by simply testing out the entire key space. The smallness of the key is dictated by the way we use it. A good cipher is expected to exhibit the *avalanche effect:* the property that ensures that switching one bit in the key would dramatically change the generated ciphertext. DES, AES, RSA and virtually all prevailing ciphers exhibit a strong avalanche effect, which has been considered a critical attraction point for every encryption scheme.

The avalanche effect is an effective defense against the Bayes attack strategy in which one learns from their failures how to get closer to their success. The avalanche effect hides the proximity of a tested key to the right key. Having the correct value of 255 bits in a 256 bits key would decrypt an AES ciphertext to a plaintext that is not any way close to the right version. And thus, the cryptanalyst has no clue how close they are. This desired effect is achieved through complex engagement of all the bits of the key with the plaintext or the ciphertext. And that is why the key must be small and of protocol dictated size. The required complexity also imposes the exponential rise in computational burden for any choice of larger keys.

When we come, as we do now, to re-imagine the role of the key, we come to realize that as attractive as the avalanche defense is, the price we pay is too high. It is the avalanche effect that invites the shadow of stealth cryptanalysis. Can we generate security without it?

We propose to replace the avalanche defense with open-ended secret size key. To say: instead of using the complexity defense, we opt for the ignorance defense.

We achieve ignorance defense by arbitrarily choosing a subset of the key which is of unknown size whenever a plaintext is brought on for encryption. Let's say the cryptanalyst has spotted the exact key for a given encryption case (instance). If this happens with an avalanche defense cipher then game is over. The same key will hold for all other encryption cases. Albeit,

for an ignorance defense cipher this extracted key may be totally useless for any subsequent encryption case because it might never be used again. And even if the right key was found for a long series of past encryption cases, the next key still remains unknown to the cryptanalyst. When past keys are extracted, the cryptanalyst will set up to build the "key puzzle" -- the key that contains all the used sub-keys. This puzzle work is (i) never conclusive because the key is of unknown size, and (ii) it is an effort applied to the key and the way it is structured. In other words, the non-avalanche defense cipher shifts the battle from the encryption algorithm to the structure of the key, where randomness reigns supreme.

We conclude thus, that the time-honored avalanche defense requirement has blinded cryptography to the viability of avalanche-non-compliant ciphers, and now this veil is removed from our eyes.

Disposing of the avalanche-defense requirement, we now re-imagine the roll of randomness in cryptography going for an encryption procedure for which the computational burden is not prohibitively affected by the size of the key. Such a "*key friendly*" encryption algorithm will allow one to use a key of any size -- large as desired, without an overbearing computational burden.

If the key we use is arbitrarily large, then we wish to make its size an integral part of the secret. Let us further denote a "*partial key encryptor*" (PKE) as an encryption procedure that may use just part of the key for any instance (case) of encryption. A PKE will protect its user from brute force cryptanalysis since the cryptanalyzed key K', as stated, may fit all the known t cases of plaintext-ciphertext, but all these cases used only a fraction $K_t \subset K$ with an unused portion ready to encrypt new messages for which K_t will not do the job.

Let M be the sum total of message material processed by a PKE. If $M < |K|$ then M cannot be the source for defining K. And since |K| -- the size of the key -- is secret, a cryptanalyst never knows if they are under the Vernam-Shannon threshold which states that as long as $|K| \leq M$,

there are ciphers that project mathematical security. And once we use a key-friendly cipher we can claim mathematical security for use limited by the size of the key.

Say then that if we use a 350 gigabyte size key on a thumb stick we can encrypt the entire Encyclopedia Britannica with pure mathematical secrecy. We just need to find a convenient key-friendly cipher.

2.2 The Ciphertext

For obscurity purposes ciphertexts are made to appear random and pattern hiding. They are the input to the procedures applied by the cryptanalyst who is hammering them to extract the respective plaintext. This suggests a means to construct a cryptanalytic barrier: inflating the ciphertext. A nominal ciphertext, C, is comprising content-bearing bits. Suppose we can mix the C bits with content-devoid bits, D. The D bits are randomized so that they look like C. We then create a rule-based combined (inflated) C+D ciphertext: C* = C (+) D. Let's assume we found a way to do so such that the intended reader of the ciphertext will readily separate C* to C and D. The recipient will discard D and decrypt C. The cryptanalyst, on the other hand, will have to regard all the bits in C* as content bearing, and apply to the inflated ciphertext the full cryptanalytic treatment. We denote such inflation-ready ciphers a 'decoy tolerant': they admit decoy bits that obfuscate the cryptanalyst and pose no burden on the intended reader.

Given a key as large as desired, and inflated ciphertext as long as wished, there rises the probability that some sections of the inflated ciphertext can be matched with assumed values within the key, such that the two elements will correspond to decrypting these sections of the inflated ciphertext with the selected elements of the key to decrypt into a corresponding plaintext that will be very plausible, according to the circumstances, misleading the cryptanalyst to a sense of baseless success, since the extracted plaintext is a result of random coincidence. With ever growing inflated ciphertext and ever larger assumed key, there will appear more and more

plausible messages extracted from the data. Among them may be the real one, but it would remain indistinguishable.

In total, we look at a cipher that may operate with a large enough key and may involve sufficiently inflated ciphertext to project any desired security up to mathematical security.

2.2.1 Parallel Encryption

The terms content-devoid used to describe the added on 'meaningless' bits, is to be interpreted with respect to the key held by the recipient of the messages. The distinction between content-bearing and content-devoid is spotted via the key, namely the content-devoid ciphertext bits appear 'content devoid' to one accessing them with the key k. The bits that look content-devoid to the intended recipient using key k may in fact be contents-bearing for a different reader of the same ciphertext, accessing it with key, k' such that $k' \neq k$.

What follows from the above is that a transmitter may encrypt a message m with key k to generate ciphertext c, and in parallel encrypt a message m' with a key $k' \neq k$, to generate ciphertext c'. When both ciphers are decoy tolerant the transmitter would be able to mix c with c' to generate c* = c (+) c', where (+) indicates mixing according to some relevant rules. When c* is approached by a recipient equipped with key k, they will regard c' as content-devoid decoy bits, discard them and decrypt c to m. In parallel a recipient of c* using key k' will regard c as content-devoid decoy bits, discard them and decrypt c' to m'.

This construction can readily extend to any n number of messages $m_1, m_2, \ldots m_n$, each using a respective key $k_1, k_2, \ldots k_n$., generating the respective ciphertexts $c_1, c_2, \ldots c_n$. Applying the proper mixing rules, these n ciphertexts will be assembled into a single ciphertext c_0. Each recipient using key k_i treating c_0 will discard all the ciphertexts $c_1, c_2, \ldots c_{i-1}, c_{i+1}, \ldots c_n$, and decrypt c_i to m_i. We have here a single ciphertext interpreted differently by different keys --

reminiscent of the Indian story about the seven blind men touching an elephant and reaching different conclusions. Numerous applications may be envisioned from such parallel encryption.

The n keys k_1, k_2, ... k_n, may be combined to a shared key, k_0 and given to the recipient. With k_0 at their disposal the recipient will be able to extract all n messages from the combined C.

The concept of parallel encryption can be used to generate a perfect secrecy cipher. In a given circumstance there are n plausible messages m_1, ... m_n that have been identified a-priori to have possibly been encrypted into the captured ciphertext. The transmitter, sharing key k_i with the recipient (i=1,2,..n) will encrypt the secret message as p_i, using k_i to generate c_i. Then the transmitter will use the other n-1 keys to encrypt all the other n-1 plausible messages, and mix the respective ciphertexts with c_i to generate c_0. The cryptanalyst is likely to spot some key k_j for $j \neq i$ and falsely conclude that the ciphertext decrypts to p_j. An omnipotent cryptanalyst will spot all the n keys, and will have to conclude that each of the n plausible message could have been the one sent. This will keep the cryptanalyst at the same position he held a-priori. Namely knowledge of the ciphertext c_0 has no contribution, which is what Claude Shannon defines as perfect secrecy [25, 26]. This pro-active upgrade to perfect secrecy may be approached through larger and larger keys and greater and greater ciphertext inflations to allow randomness to offer increased probabilities for false plaintexts to mislead the cryptanalyst.

2.3 The Encryption Procedure

Last source for randomness to be examined is the procedure used to build the ciphertext from the plaintext with the help of the key. One can devise different procedures which admit various ways for randomness to play a role. For example, a long string of plaintext bits may be randomly spliced to substrings, each of these substrings is separately encrypted. The respective ciphertexts are concatenated to a single ciphertext. The ciphertext is subsequently spliced and decrypted. The various randomized slices will create a different ciphertext every time the same plaintext string will be given for encryption.

There are various algorithms that generate different ciphertext each, while all these ciphertexts decrypt to the same plaintext.

One important class of randomized decisions is the decisions that select which part of the key to use for any instance of encryption. Let the communicators use a key k of size |k|=s bits. A key selection randomizer will pick q certain bits to serve as key material for a pending encryption computation. The maximum number of options (o) for picking q=1,2,...s key bits is:

$$o = \Sigma\, s!/(s-q)!$$

which is an astronomical number even for small size keys. The cipher will have to be able to communicate to the recipient the identity of the selected q, and do so while keeping the cryptanalyst in the dark.

The more randomized decisions intervene in the process of generating the ciphertext from the plaintext, the more the data is clear of any sustainable pattern, and the more it limits the cryptanalyst to brute force attacks. Such attacks are never conclusive because of the indefinite size of the key.

2.4 Combining Randomness

The shared randomness is only the key. The rest is unilateral randomness. The transmitter governs the use of randomness to specify the encryption procedure, to select a part of the key, and to inflate the ciphertext to the desired degree. The quality of such "Tesla cipher" as these ciphers may be called, will be appraised by the degree of randomness involved in their operation. A fully operational Tesla Cipher, or "Tesla" for short, will wipe out any pattern, and channel the attacker to brute force attacks that are never conclusive since the size of the key remains a secret, and no amount of use will reveal it.

The randomized decisions taken within the Tesla cipher can be guided to present a matching response to the prevailing and dynamic threat. Probability teaches that cutting the plaintext to smaller slices, and applying fresh randomness to each slice, increases the randomness projected before the cryptanalyst. Selecting larger key slices to encrypt a given plaintext will also increase the cryptanalytic barrier (more key material to unearth). Greater inflation of ciphertexts will increase the chance for the cryptanalyst to extract a false plaintext and regard it as the true plaintext.

Most if not all of these increased security selections involve a more "messy" process, handling of more data, communicating more bits. Such burden should be considered as the price for security. What is critical here is that the user is the one who decides how much inconvenience to put up with in order to project the desired security that matches the perceived threat.

The Tesla cipher may be built with an automatic threat response: logic that impacts the randomized selections according to the perceived threat. The cipher will detect attempts to breach it, as opposed to normal use, and respond accordingly -- automatically. For example, if the same plaintext is fed in time and again, it suggests cryptanalysis, and in response the Tesla cipher will lean towards randomized selections that increase security like smaller slices of plaintexts, larger key slices, and more inflation of the ciphertext. The same in response to repeat attempts of very short plaintexts, etc.

The transmitter may have an override and guide the range for randomized decisions according to their will.

2.5 Defining a Tesla Cipher

The following properties will define a cipher regarded as a "Tesla Cipher". The name honors both Nikola Tesla and the Tesla EV both envisioned a future different from the linear projection of the past.

1. A Tesla cipher projects security without reliance on mathematical complexity.

2. A Tesla cipher is a cipher that builds a cryptanalytic barrier around the generated ciphertext on the basis of both shared and unilateral randomness such that the barrier is greater the more randomness is deployed.

3. A Tesla cipher uses a secret size key that can grow without a limit on its size and with no prohibitive impact on the nominal computational burden of the encryption and the decryption procedures.

4. A full-fledged Tesla cipher will be decoy tolerant, namely its content-bearing ciphertext bits can be mixed with content-devoid bits such that the intended reader will be able to distinguish between the content-bearing bits and the content-devoid bits and decrypt only the former while an attacker will not be able to make this distinction.

5. A full-fledged Tesla cipher will be a "Trans-Vernam Cipher" (TVC), namely a cipher for which the key can be made sufficiently large and in combination of a sufficiently diluted ciphertext will achieve greater than Vernam (mathematical) security in as much that knowledge of the ciphertext does leave both the content and the size of the plaintext undetermined between 1 bit and the size of the ciphertext.

6. A Tesla cipher can operate with one key shared by two or more communicators and do so without key status monitoring. Hence the Vernam cipher is not a Trans-Vernam cipher.

3.0 Implementation Considerations

The unique challenges of a Tesla cipher are (i) sharing the large key, (ii) communicating the inflated ciphertext, (iii) storing the large key. The first challenge poses the greatest concern. The best way to share a Tesla key is to do so offline over a flash drive or comparable device. Sharing over a line secured by mathematical complexity undermines the Tesla advantage. However,

randomness-based sharing options [19] and advancement in quantum sharing offer a satisfactory solution.

Tesla can be implemented in a nominal software mode, or over a latched key that is removed before and after use. It can also be implemented as a full system locked in a secure enclosure [23, 24] . The enclosure will contain the key, the software, and the source of randomness. It will admit a plaintext and generate a respective ciphertext and vice versa. The packed-in ad-hoc randomness will keep the content of the key from being exhaustively deduced by a user who does not defeat the security of the enclosure.

3.1 Post Quantum Cryptography

The US National Institute for Science and Technology (NIST) is leading a global campaign to generate enhanced complexity algorithms to frustrate the emerging quantum computing cryptanalysis. The 'shadow' discussed herein applies much the same to the NIST quantum defense. Already several selected NIST candidates have been breached, and that is only based on the known attack algorithms, which were published last century. It is most likely that the coming attack will be based on advanced algorithms developed behind veils of secrecy, and against which the so called post quantum ciphers have not even been designed, let along tested. It is a situation which casts most favorable light on the Tesla alternative. [3,5]

3.1.1 Lifeboats on the Titanic

The Titanic was the technological marvel of the day, but when it was painfully surprised by unexpected failure scenario the only way to save lives was to deploy low-tech devices, lifeboats. This occurrence may serve as a guide for today's cyber security executives. The power of inertia is enormous. However persuasive this and similar writings are, people will go on doing tomorrow what they have been doing all along. No one expects AES-256 to be retired soon.

Moreover, it is fully expected that the emerging harvest of post quantum (math based) ciphers emerging from the US National Institute of Standards and Technology will be embraced, adopted and deployed against the approaching threat of quantum computers. So be it, but let us recall the Life Boats on the Titanic. Compare them to Tesla, if you will. Tesla uses primitive mathematics, the lifeboats were powered by oars... In operative terms, a prudent CIO should deploy Tesla for the eventuality that the quantum attack comes stronger than expected and the NIST post quantum defense performs less than hoped for. In that case, the only thing between operational continuity and a cyber catastrophe is a Ready-to-Go Tesla.

3.2 AI Resistance Cryptography

Artificial Intelligence introduced a new attack vector wielded by modem cryptanalyst: using AI power to develop a list of highly likely plaintext messages that could have been used in a particular situation. Having such a list to be casted against a captured ciphertext will allow the cryptanalyst to sort out candidates for which the likelihood of finding a key to match the ciphertext is small. This finding will diminish the entropy of the ciphertext and will lead a way to a breach. This is especially alarming for cases where the plaintext is a choice among a well-known set of options. Many automotive devices make decisions or get instructions which are a particular selection from a known set. These cases are exceedingly vulnerable to AI attack. Considering this threat, Stela looks particularly attractive because it might keep the entropy high among all the possible plaintexts by using a sufficiently large key and sufficiently diluted ciphertext. See more [4]

3.3 Battery Friendly

Tesla ciphers are free from complicated computations and use only basic bit operations. Security is achieved through large keys which are stored with no power consumption and require

little power to access and even to communicate. This property renders Tesla ciphers to be battery friendly. In the variety of applications where ciphers are powered up by drainable batteries, this attribute is a critical advantage.

3.4 Sources of Randomness

Currently cryptography is using mostly algorithmic randomness, of which John von Neumann remarked that its user does not understand neither algorithms, nor randomness. It would make little sense for Tesla users to rely on algorithmic randomness, which will pull them back to be vulnerable to stealth mathematical cryptanalysis.

At a minimum, one would use filtered algorithmic randomness. Such is generated by passing algorithmic randomness through a filter that discards sections that do not pass a randomness test. A complying filter has been designed [17]. It measures subsections of a randomness flow according to the level of symmetry detected in such substrings. The more symmetrical a string, the less random it is. The threshold of acceptance can be set at will and thereby the algorithmic predictability is shuttered.

Next in line are physical complexity devices, which are in theory deterministic but owing to their immense complexity they may be properly regarded as high-quality randomness. A recent patent [15] is based in fluctuating electrical current across fluid in which bubbles rise and change the conductivity in the circuit.

Quantum randomness is claimed by physics today to be perfectly random by any measure. There are easy ways to capture quantum randomness. The oldest one is based on fluctuations in radioactive decay of a rather stable source. A new commercially available source is based on tracking photons pinged on a slanted half mirror. They have a 50% chance to go through the half mirror and 50% chance to bounce off [18].

Quantum grade randomness can also come in a prepackaged form. Such is the “Rock of Randomness” technology [16] that prepares a lump of matter comprised of material ingredients which differ greatly in their electrical conductivity. These ingredients are arranged in randomized patches. Measuring lump conductivity between various lump points provides a rich supply of random readings. The Rock of Randomness can be manufactured in a mini format so it can fit neatly in a small enclosure and serve as a randomness source used in a self-contained Tesla cipher.

3.5 Socio Political Implications

Until now the public was riding on trains and buses, so to speak. You could choose the vehicle but not the speed. The train operator controlled how fast the train goes. Similarly the cipher designer determined the security level. The user could select among available ciphers. Looking forward the public moves about in private cars, extra maintenance work, but freedom to push the gas pedal to the desired degree. The Tesla user gets the responsibility and the power to decide how much security to project for the communication payload they are engaged in.

Randomness is plentiful (the equivalent for gas), and ordinary communicators can deploy enough of it to ensure their privacy against anyone. Today’s prevailing hierarchy is enduring no more. Much as in the allegory of the one dice versus two dice, the 'dumb' user will be able to defend his data assets against the smartest and best equipped cryptanalyst. This new constellation of power is bound to have a meaningful socio-political impact, which is best analyzed by professionals in these matters.

3.6 Samples of Tesla Ciphers

Tesla ciphers come in various flavors. Their development has just begun. Presenting three samples.

3.6.1 Trip on Key (ToK)

Imagine a map (graph) : a bunch of locations (vertices) connected through roads (edges). A trip of this map can be described by listing the visited locations, or by listing the traversed roads – same trip. Anyone holding the map can change the description from location-sequence to road-sequence and vice versa. Hence by treating the map as a shared cryptographic key, the location-sequence can be viewed as a plaintext while the road-sequence be regarded as ciphertext. The structure and the size of the map are secret, so an attacker does not know whether the next message will mark a trip on a section of the map that was not traversed before. Hence no conclusive brute-force cryptanalysis is possible. Encryption and decryption amount to simply reading the opposite sequence from the map. It is a most simple computing action which is clearly independent of the size of the map. [9, 13].

3.6.2 BitFlip

Let each letter a_i of an alphabet A comprising n letters a_1, a_2, …. a_n be associated with a unique bit string s_i of an arbitrary size, 2h. This defines a *2hn* size key. To transmit letter a_i, one sends over a pointer string t_{ij} such that the Hamming distance between s and t_{ij} is *h*. There are $p = (2h)!/(h!*h!)$ such strings: j=1,2,…p. Each letter in A can be represented by an arbitrary (secret) number of strings, which have not used before. By keeping *h* large enough, the chance for a false match of ciphertext and key candidates grows – misleading the cryptanalyst. By sending pointer strings that point to no letter in A, or point to two or more – the decoy effect is achieved. [7, 8]

3.6.3 The Unary Cipher

The Unary cipher translates a plaintext into a unary language, adjusting it to become a very large binary string of equal number of ones and zeros, and then using a large enough key to ensure full range of permutation for that string. The transposed string (the ciphertext), is de-transposed by the recipient then converted back from unary language to ASCII. It can be shown that there are different de-transpositions that lead to plausible, but false plaintexts. The larger that transposition key, the greater the security [10]

4.0 The Innovation Solution Protocol

The search for an alternative for mathematical complexity was conducted according to the guidance of the Innovation Solution Protocol [28, 29]. Work done in the area of information representation was abstracted to include cryptography. Extending the data representation issue to encryption has shown a gap between the prevailing ciphers and the new method which was based on the fact that a pathway on a network may be expressed either by the sequence of visited vertices or by the sequence of moved on connectors between these vertices. In analyzing, then abstracting the difference between the two approaches, it became clear that the new method is based on key size not on mathematical complexity. And it further became clear that key material can be added by the user, while math-based cryptography offered a fixed measure of security. The rest unfolded. All the way, the Innovation Solution Protocol provided the innovation evolution map, and the resource estimates needed to appraise the innovation load ahead.

5.0 Summary and outlook

The starting point of this writing was the alarming realization that math-complexity cryptography operates under an irremovable shadow of stealth cryptanalysis. This shadow was deemed unacceptable in a situation where humanity migrates to cyberspace, and privacy and security are nonnegotiable demands. This spurs a search for a different basis for projecting security. Randomness was identified as a suitable alternative. This article further outlines ways how to use randomness as a replacement for mathematical complexity.

The Tesla analogy was deemed fitting for this proposal. Elon Musk pioneered the pivoting from crude oil as basis for transportation practice. We propose pivoting from math complexity as a basis for cyber security practice.

6.0 Reference

The underlying idea for Tesla Ciphers is pattern devoid cryptography. The following is a review article discussing the features of randomness based security. It is rich with references:

1. G. Samid "Pattern Devoid Cryptography" https://eprint.iacr.org/2021/1510

2. This discussion also is featured as the appendix published with the related novel (thriller) "The Cipher Who Came in From the Cold" HTTPS://BitMintalk.com/thriller

Further features for pattern-devoid cryptography are discussed in subsequent articles:

3. G. Samid "The Prospect of a New Cryptography: Extensive use of non-algorithmic randomness competes with mathematical complexity" https://eprint.iacr.org/2023/383

4. G. Samid "AI Resistant (AIR) Cryptography" https://eprint.iacr.org/2023/524

5. QSEC https://www.bitmintalk.com/qsec

A broad discussion on randomness is featured here:

6. G. Samid "Randomness Rising: The Decisive Resource in the Emerging Cyber Reality" https://www.bitmint.com/RandomnessRising_GSamid_H1o16.pdf

Various Tesla Ciphers are also published:

7. G. Samid, S. Popov "BitFlip: A Randomness-Rich Cipher" https://eprint.iacr.org/2017/366

8. G. Samid "Threat-Adjusting Security: BitFlip as an AI-Ready, Post-Quantum cipher" https://eprint.iacr.org/2018/084

9. G. Samid "Drone Targeted Cryptography" ICOMP'16 - The 17th International Conference on Internet Computing and Internet of Things * https://eprint.iacr.org/2016/499

10. G. Samid, "A Unary Cipher with Advantages over the Vernam Cipher"

11. USPTO: "SpaceFlip Plus: Ordinal Cryptography" 7-22-2021 Publication number: 20210226923

12. US Patent "Transposition Encryption Alphabet Method (TEAM)" Patent # 11038668

13. US patent "Bitmap Lattice: A Cyber Tool Comprised Of Geometric Construction" US patent 10911215

14. USPTO "SpaceFlip: Unbound Geometry Security" Patent number: 10790977

Methods to generate non-algorithmic randomness:

15. USPTO: "RandoSol: Randomness Solutions" Publication number: 20210075593

16. USPTO "Rock of Randomness" Patent number: 10467522

17. Samid, G. "Randomness as Absence of Symmetry" The 17th International Conference On Information & Knowledge Engineering (Ike'18: July 30 - August 2, 2018, Las Vegas, Usa)

18. IDQ https://www.idquantique.com/

Complexity Devoid Privacy generators between online Strangers:

19. USPTO: "Randomized bilateral trust (RABiT): trust building connectivity for cyber space" Patent number: 10798065

Applications for Tesla Ciphers:

20. USPTO "Effective Concealment of Communication Pattern (BitGrey, BitLoop)" Patent number: 10608814

21. USPTO "Cyber companion: attaching a secondary message to a primary one" US Patent 10541954

22. USPTO "Skeleton network: physical corner stone for the towering cyber house" Patent number: 10523642

Secure Enclosures for self-sustaining Tesla cipher:

23. USPTO "Digital Transactional Procedures And Implements" US Patent 10,445,730

24. USPTO "Hard Wallet: A New Trust Basis For Digital Payment" US Patent 11,062,279

Background Items:

25. Shannon Proof of Vernam's Cipher Unbreakability
https://www.youtube.com/watch?v=cVsLW1WddVI&t=135s

26. "Communication Theory of Secrecy Systems". Claude Shannon (1949)
http://netlab.cs.ucla.edu/wiki/files/shannon1949.pdf

27. USPTO "Secret Signaling System" US1310719A (The Vernam Cipher)

The Tesla Cryptography was processed using the tools of the Innovation Solution Protocol, and AI Assisted Innovation:

28. The Innovation Solution Protocol: www.InnovationSP.net

29. Samid, G. Artificial Intelligence Assisted Innovation
https://www.intechopen.com/chapters/75159

TABLE OF CONTENTS