PAST KNOWLEDGE PEOPLE

ISSN: 3066-1730

Research Article

Journal of Electrical and Computational Innovations

Transmitting Secrets by Transmitting only Plaintext: *Hiding-in-Plain-Sight Cryptography*

Gideon Samid*

Electrical, Computer and System Engineering, Computer and Data Sciences, Case Western Reserve University, United States

***Corresponding Author**
Gideon Samid, Electrical, Computer and System Engineering, Computer and Data Sciences, Case Western Reserve University, United States.

Submitted: 2025, Nov 22; **Accepted:** 2025, Dec 24; **Published:** 2026, Jan 20

Citation: Samid, G. (2026). Transmitting Secrets by Transmitting only Plaintext: Hiding-in-Plain-Sight Cryptography. *J Electr Comput Innov, 3*(1), 01-08.

Abstract
Presenting a novel use of encryption, not for hiding a secret, but for marking letters. Given a 2n letters plaintext, the transmitter encrypts the first n letters with key K1 to generate corresponding n cipherletters, and encrypts the second n letters with key K2 to generate n corresponding cipherletters. The transmitter sends the 2n cipherletters along with the keys, K1 and K2. The recipient (and any interceptor) will readily decrypt the 2n cipherletters to the original plaintext. This makes the above procedure equivalent to sending out the plaintext. So why bother? When decrypting the 2n cipherletters one will make a note of how the letters that were encrypted with K1 are mixed with the letters encrypted with K2 while keeping the original order of the letters encrypted with each key. There are 2^n possible mixings. Which means that the choice of mixing order can deliver a secret message, S, comprising n bits. So, while on the surface a given plaintext is sent out from transmitter to recipient, this plaintext hides a secret. Imagine a text messaging platform that uses this protocol. An adversary will not know which plain innocent message harbors a secret message. This allows residents of cyberspace to communicate secrets without exposing the fact that they communicated a secret. Expect a big impact on the level of cyberspace privacy.

Keywords: Communication Privacy, Stealth Cryptography, Plain Sight Cryptography

Preamble

One of the curious findings of "AI Assisted Innovation" is that human innovation is inertia bound [1]. Innovation that fits in the flow of prior innovative steps is quickly adopted, often without sufficient scrutiny, while innovation that calls for a change of direction faces soft rejection, rejection that hinges on style, on source, on vague arguments like "it's not persuasive" etc. The presentation herein is a case in point. It breaks away from the premise that encryption is used to hide information. Here it is used to mark information, not to hide it. It also breaks away from the premise that a ciphertext commits to its generating plaintext. The ciphertext herein decrypts to one message using one key, and decrypts to another message using a second key. Digital steganography is commonly practiced through hard-to-find add-on bits. Herein the pre transmission data (plaintext) and the post transmission data (plaintext) leave no clue of the steganographic message that passed from transmitter to recipient. It appeared in the data in motion only, and not with any add-on bits, just by reordering of letters encrypted with different keys.

1. Introduction

The ultimate privacy in communication is when a group of communicators can have a conversation, and not its contents, neither its occurrence is exposed against the will of the communicators. Proposing to achieve this state of ultimate privacy (UP) by allowing communicators to converse in the open (clear conversation) wherein the same communication encompasses a hidden conversation that does not expose its occurrence. This level of ultimate privacy is achieved to the extent that the clear conversation between the communicators by itself is not pointing to a hidden exchange. That means, if Alice and Bob have normal business conversation between them, they can hide in its exchange a hidden message -- hidden in plain sight (HIPS) because there is no more than the clear exchange that runs between Alice and Bob, only that this clear exchange is so set up that it carries a hidden exchange. The efficacy of HIPS cryptography is hinged on (i) the extent that the clear conversation is not attracting scrutiny, or at least can be credibly denied as having any other purpose except what is evident from the nature and timing of the clear conversation, and (ii) on the extent to which the hidden conversation is so well hidden that an examiner of the clear conversation will find no evidence to the existence of the hidden message.

We will describe how to handle the first and the second challenge above. The first challenge is conveniently handled the normal exchange between the collaborators. Normally collaboratros exchange information (that is not secret), this exchange will well qualify as the carrier clear text to be loaded with the secret message (payload). Today it is easy to employ any AI tool to exchange innocent looking text between two communicators. In that case the message in the clear text may be of no interest, it is only a carrier for the payload.

It is important to note that the content of the cleartext has no bearing on the payload (the secret). The payload is handled through nominal ("fake") encryption. Encryption procedure is used over the cleartext to carry the payload. It is "fake" because the ciphertext is delivered together with the keys needed to decrypt it, which is what the recipient does (as well as any hacker along the way). A more civilized term will be Plaintext-to-Plaintext cryptography (P2P).

Spartacus: In the movie "Spartacus", based on historical records, the Romans capturing the rebellions are trying to spot their leader, Spartacus. When they ask "Who is Spartacus?" all the rebels reply in unison "I am Spartacus!" keeping the Romans baffled. Hidden-in-Plain-Sight, HIPS, works the same. The payload is added to normal and proliferating communications without standing out, compelling an attacker to suspect any and all communications. Given that hidden secrets are a very tiny fraction within the flood of Internet traffic, emails, messages, downloads, etc, this indistinction is a very effective tool, and a great contributor to privacy.

2. Plaintext to Plaintext (P2P) Cryptography

This Hidden-in-Plain-Sight, HIPS, method, P2P, is based on exploiting the flexibility built in into decoy tolerant ciphers.

A decoy tolerant cipher will distinguish between (i) ciphertext material that bears content and decrypts to its generating plaintext, and (ii) noise -- ciphertext material that is decrypting to no valid plaintext (per the prevailing key), and hence is to be ignored. The most commonly used decoy tolerant cipher is BitFlip. A host of such ciphers is described in the chapter "Pattern Devoid Cryptography" [2].

BitFlip: the BitFlip cipher operates on an alphabet A comprising a letters where each letter is a randomized bit string of size 2t bits. A 2t-bits size string c that has a t-bits Hamming distance from a plaintext letter, points to it. A 2t-bits string that points to no plaintext letter is decoy. What is regarded as decoy per a key K will be a valid ciphertext letter per a different key K'.

Alice and Bob wish to establish a HIPS channel. To that aim they establish an open communication channel using a HIPS compliant text processor. They record a high level of open communication, then when the need arises they send to each other a secret payload. The payload itself may be encrypted through a decoy tolerant cipher so the reader can readily establish whether the extracted payload candidate is a payload indeed or empty randomness.

Whenever the HIPS processor is used plainly without injecting a payload into it, then the construction of the ciphertext string should be done randomly to confuse the attacker as to whether it hides a payload or not.

An attacker monitoring Alice and Bob reading their open exchange will have no grounds to suspect that a secret message is hiding in plain sight. There is no other secret communication between Alice and Bob, everything they say to each other is through the HIPS processors. And if there is a suspicion based on some external circumstances then it cannot be substantiated.

Broadcasting: Alice and Bob can communicate through HIPS in a broadcasting mode.

Alice broadcasts a blog, a message board, a website content -- using HIPS processors. The cyberspace public is downloading, reviewing, interpreting the HIPS packages and for most of the readers there is nothing more than what the plain broadcast message says.

For Bob though that podcast is regarded as "armed communication", containing the clear text (the plain message) and hiding a shared secret between him and Alice. The hiding is through writing and interpreting the particular order of letters in the ciphertext.

This way Bob will be receiving messages from Alice. Since Bob does what so many online surfers do -- download Alice's podcast, there is no indication that Bob is the target of the HIPS secret. Bob in turn may either send Alice messages, or to be more obscure Bob will broadcast his own podcast which many in cyberspace will download -- including Alice. Bob will inject his payload into his podcast content and thereby send messages to Alice.

In summary, with both Alice and Bob broadcasting to the world, and both downloading each other broadcast, the two can communicate in a way which is hidden in plain sight. There is no indication that they are talking with each other because the podcasts they put forth are being used and downloaded by many others in cyberspace. The HIPS aim is achieved -- the communication is properly hidden.

Keys Visibility: In the basic deployment the keys of the decoy tolerant ciphers are packed into the ciphertext to allow every one encountering the package to decrypt the messages into their original plaintext. However, this can be changed. The keys can be withheld -- some or all, from one, few or all of the intended recipients, thereby security can be managed.

Steganography: Comparison

Steganography as commonly practiced today is less systematic and more particular than the HIPS concept presented here [3-22]. Most methods rely on video and audio as message carrier, which is not as handy and as common as text. Text based steganography is mostly based on format and appearance (font type, size, location on paper), which requires text in a very limited environment. Contextual text methods (e.g. the first letter of each word is part of a secret message) require dedicated text - a burden. With HIPS, every body of text is a good carrier, no modifications needed. The payload is handled through a 'fake encryption' protocol over the clear text, where it is regarded as 'fake' because it is de facto plaintext-to-plaintext encryption, the encryption protocol is used to upload the payload and deliver its steganographic mission. Normal textual exchange is the most common information exchange, creating a big hiding environment for the HIPS secret messages.

3. Methodology

Hidden in Plain Sight (HIPS) cryptography is essentially

1. A cryptographic method called "Hiding in Plain Sight", HIPS, used by a transmitter and a message recipient where both are remotely connected over cyberspace, and wherein a non-secretive text, "clear text", contains a secret message called "payload" (π) and where being clear text, it draws no attention to the payload, thereby allowing for transmission of secret messages where neither the content, nor the fact of the transmission is exposed to an adversary;

HIPS operates as follows:

Let M be a clear text message comprising 2n letters of a given alphabet α, let M_1 be the message written as the first n letters in M, and let M_2 be the message written as the last n letters in M.

Let DTC be a "Decoy Tolerant Cipher" which is a cipher operating over α, through a key K, and that distinguishes between (i) a ciphertext letter that is to be decrypted to its generating plaintext letter, and (ii) a decoy ciphertext letter which does not decrypt to any letter in α when decrypt-processed with key K.

Let M_1 be DTC-encrypted with K_1 to the corresponding ciphertext C_1 comprising n ciphertext letters, each by order decrypts to its corresponding letter in M1.

Let M_2 be DTC-encrypted with K_2 to the corresponding ciphertext C2 comprising n ciphertext letters, each by order decrypts to its corresponding letter in M_2.

Let the payload be written as a bit string containing n bits; the transmitter builds a composite ciphertext CC by concatenating individual ciphertext letters from C_1 and C_2, as follows:
defining:
(i) π_i as the i-th bit in π,
(ii) c_{1i} as the i-th letter in C_1
(iii) c_{2i} as the i-th letter in C_2
(iv) cc_i as the i-th letter in CC

Constructing CC by taking letters from C_1 and from C_2 according to the following rule: given the CC being constructed by moving letters from C_1 and C_2 one after the other concatenating one by one, and given a state of CC where it is constructed from q letters from C_1 and r letters from C_2, then setting the q+r+1 letter in CC to comply with:

If $\pi_{q+r+1} = 0$ ***then*** $cc_{q+r+1} = c_{1q+1}$
If $\pi_{q+r+1} = 1$ ***then*** $cc_{q+r+1} = c_{2r+1}$

for q = 1,2,.. and r =1,2,.. until q + r = n

And from that state on, CC is constructed by randomly selecting the remaining letters from C_1 and C_2, until all the letters in C_1 and C_2 have been moved to construct CC.

Preparing a ciphertext package containing CC and K_1 and K_2 , sending it to the recipient over insecure channel.

The recipient decrypting CC first via K_1 to M_1, then via K_2 to M_2, then constructing M by concatenating M_1 and M_2: $M = M_1 \| M_2$ thereby re-constructing the clear text message M, then constructing π as follows: for i = 1,2,...n

If $cc_i = c_{1j}$ ***for some*** $j = 1,2...n$ **then** $\pi_i = 0$
If $cc_i = c_{2j}$ **for some** $j = 1,2...n$ **then** $\pi_i = 1$

thereby π is constructed by the recipient which concludes a HIPS round.

2. The method in paragraph 1 wherein the payload is a ciphertext generated by a DTC from a secret plaintext, "The HIPS secret", by using a Payload-DTC key K_π.

Alternatively, the payload is decoy ciphertext that when decrypted with K_π points to no plaintext; where in the first option the communication package is regarded as "armed" and in the second option the communication package is regarded as "empty".

The Payload-DTC and K_π are shared between the recipient and the transmitter.

3. The method of paragraph 2 wherein the transmitter executes t successive HIPS rounds, most of them empty and a minority of them armed; an attacker will decrypt CC into M using K_1, K_2 which are part of the ciphertext package, but will have no indication which of the HIPS rounds is armed and which are empty.

4. The method of paragraph 1 wherein the clear text is written by the transmitter to either send to the recipient messages for which no secrecy is required, or the clear text is written to send to the recipient messages that would draw no suspicion to be hiding a secret -- look innocent -- serving the normal exchange between the communicators, when observed by an adversarial cryptanalyst, but these clear text messages only serve as a "blanket" to wrap in it the messages carried by the payloads, and their content is of no interest to the recipient.

5. The method of paragraph 1 wherein the clear text is written by an artificial intelligence, AI module that is trained in the normal communication between the transmitter and the recipient, and generates a clear text designed not to draw suspicion for a presence of hidden payload.

6. The method of paragraph 1 wherein two communicators are sending each other clear texts wherein no proper payload is used, and a random numbers generator is used to generate a fake payload, these rounds of communications render the communicators ready to use armed.

7. The method of paragraph 2 wherein normal messaging tools, email, phone messaging are operated in the HIPS mode, so a large number of the members of the text messaging public is using it, wherein the overwhelming majority of the HIPS rounds are empty, and only a small minority of the HIPS rounds are armed.

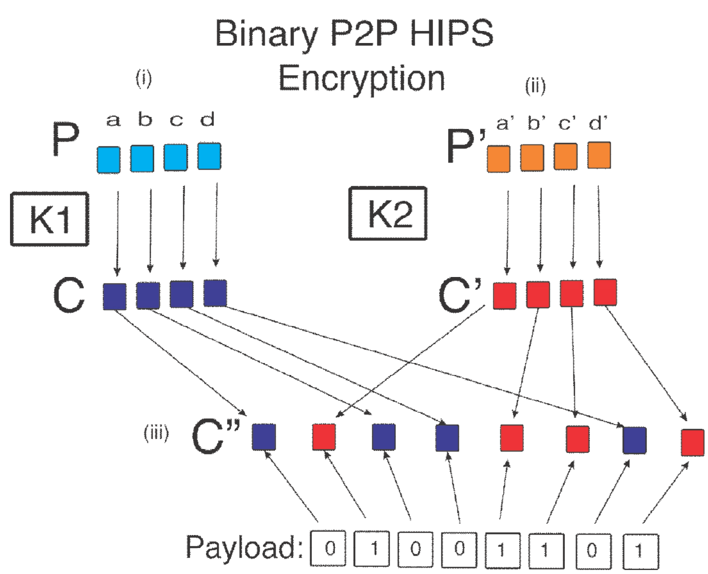
8. The method of paragraph 1 applied in a conversation mode wherein a group G of g parties
(i) share a payload-DTC key
(ii) establish an extensive cross messaging environment within G wherein they run conversations through the HIPS protocol, exchanging clear text messages that require no secrecy, and use a large plurality of empty rounds
(iii) use armed rounds in a minority of HIPS rounds within G without drawing suspicion from an observing adversary.

9. The method in paragraph 8 exercised in broadcast mode wherein party i, i = 1,2,..g broadcasts clear text i that is downloaded by a multitude of online readers, which are not in G but among them are the parties in G who share an agreed-upon Payload-DTC and a respective $K\pi$;

clear text i is comprising a majority of empty HIPS rounds, and a minority of armed rounds which the other parties in G detect and properly interpret;

party j, j = 1,2,..g. $j \neq i$ is responding to a payload sent by party i, by broadcasting clear text j that is downloaded by a multitude of online readers among them the parties in G; the parties in G properly interpret the payload from party j, thereby the group G is exercising a clear text conversation while also conducting a HIPS conversation for which neither the contents nor the fact of its occurrence is visible by other than the members of G.

4. Drawings



This figure depicts binary mode P2P (plaintext to plaintext) encryption. A plain message "abcd" is being encrypted with a decoy tolerant cipher using key K1, creating 4 ciphertext letters (darker blue), where each letter in the plaintext, P is encrypted to a different letter in the respective ciphertext (darker color). It is shown as P → C.

Figure 1: Binary P2P Encryption

This figure depicts binary mode P2P (plaintext to plaintext) encryption. A plain message "abcd" is being encrypted with a decoy tolerant cipher using key K1, creating 4 ciphertext letters (darker blue), where each letter in the plaintext, P is encrypted to a different letter in the respective ciphertext (darker color). It is shown as P → C.

In parallel another plaintext message "a'b'c'd'" is encrypted by the same cipher only using a different key K2 ≠ K1 resulting in cipher text C', comprising the same number of letters as the corresponding plaintext.

The picture shows the payload example as an 8 bits long bit string: 01001101 which the transmitter is encrypting into the composite ciphertext C" by adding the next letter from C when the corresponding bit in the payload shows 0, and adding the next ciphertext letter from C' when the corresponding bit in the payload shows 1. So, one by one the composite ciphertext is created, carrying the two open ciphertext C and C' together with the secret payload expressed through the order in which C" has been constructed.

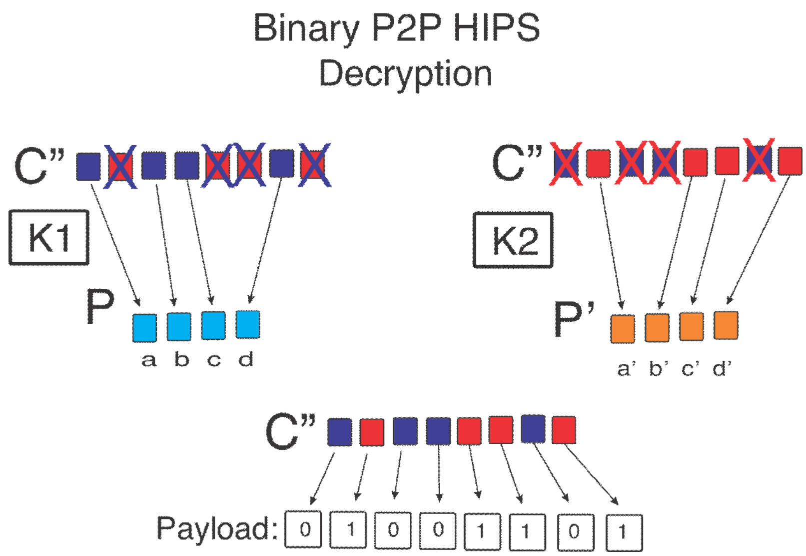


Figure 2: Binary P2P HIPS Decryption

This picture is continuation from Fig-1. It shows how C" fed into the first cipher using K1 is discarding all the K2 letters and reading the letters from C one by one decrypting them through K1 and extracting the corresponding plaintext P. It also shows how the same C" is processed through the same cipher but with key K2. Accordingly, all the K1 letters are discarded in C" and only the ciphertext letters from C' are fed to the decryptor to extract the proper plaintext message P'. (a' b'c'd')

The order of the K1 letters versus the K2 letters guides the reader to reconstruct the payload strong 01001101.

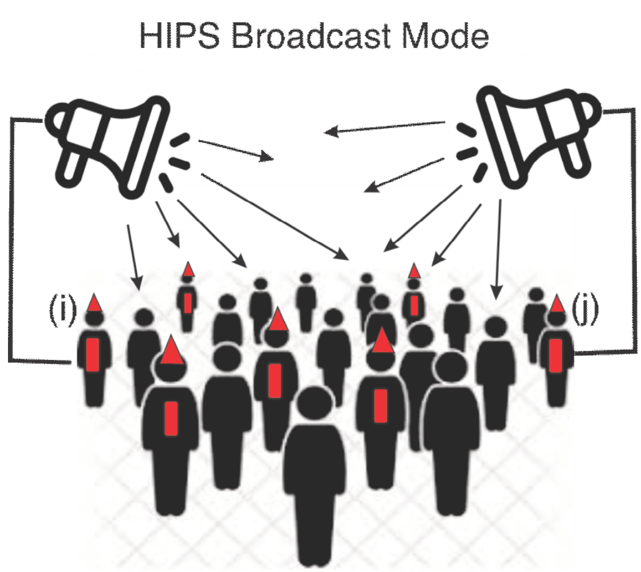


Figure 3: HIPS Broadcast Mode

This figure depicts HIPS communication in a broadcast mode. A community of online surfers is shown, among them are members of a group G, they are marked with a pointed hat and a patch on the back. The figure shows member i of G broadcasting in cyberspace via a podcast, or a public channel or website. This clear text broadcast is downloaded and brought to the attention of many members of the public including members of group G. While member i may broadcast mostly empty rounds of HIPS where the clear text is the aim of the broadcast, member i may when needed hide an armed clear text - a clear text containing a HIPS secret. The public in general will not see a difference, but members of G will notice and will properly interpret the hidden payload secrets sent out to them by member i. In response member j of G will broadcast clear text j, which too will be downloaded by members of the public, among them members of G. Should member j reply to member i, member i will well receive the reply. And so members of G conduct a secret conversation in plain sight, covered by a conversation of clear text among them.

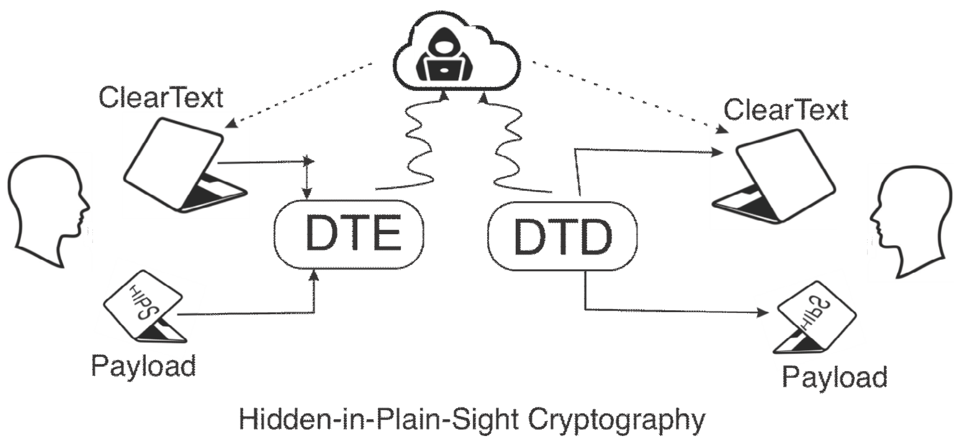


Figure 4: HIPS Configuration

This figure depicts the HIPS configuration: a DTE -- Decoy Tolerance Encryption is encrypting the clear text before releasing it through the Internet, and a DTD -- Decoy Tolerant Decryption unit decrypts it, displays the clear text and the secret payload is present. Hackers will have access, at most, to the clear text at both ends.

5. Deployment

For HIPS cryptography to work, it must be implemented through text messaging and email modules which are HIPS compliant, namely, written text is encrypted with a decoy tolerant cipher, with keys included, and the encrypted ciphertext is then decrypted on the receiving end with HIPS compliant processor decrypting the ciphertext with the keys that are included in the package. Thereby the cipher is not used to hide content rather to mark content. Specifically the order of letters encrypted by two selected keys is monitored and is interpreted as a secret message (payload). Since payloaded text is a small fraction from the overall textual traffic then the payload is protected by its environment. The net result is that Alice and Bob communicate with each other using only innocent text where the contents therein is suggesting no secret communication between them.

6. Security

The security projection of HIPS is different from nominal situations. Typically, one argues security on the grounds that no breach algorithm was published, and the cipher designers believe that the adversary will not surprise them with unexpected mathematical and computational capability. Anyone recalling the story of Alan Turing and the German Enigma cipher will be uneasy about this security proclamation. Yet, articles are being published, and authorities issue certificates for such cryptographic solutions that suffer from this "Single Smart Mathematician" syndrome.

HIPS is a category apart; it builds its security on abundance of randomness.

We present our analysis as follows: Imagine a situation where the environment of interest includes two remote parties Alice and Bob. They practice HIPS. The adversary suspects they do, and analyzes the innocent text they send each other. Since the keys are attached to the ciphertext, the adversary will read the order in which the letters encrypted with key K1 and mixed with the letters encrypted with key K2, as the procedure dictates, this mixing will be translated to a bit string S, containing n bits (per each Alice-Bob plaintext message containing 2n letters). S was constructed from a so-called payload cipher, Cp, using a payload key Kp. The only necessary requirement from Cp is that it generates a random-looking ciphertext. So, a payload message Mp, encrypted with Cp using key Kp generates the string S which Alice passed to Bob by sending him the 2n letters plaintext. Since the adversary suspects Alice and Bob to be using HIPS, the adversary will apply cryptanalysis measures to crack S and extract the payload Mp.

In the formal protocol for HIPS we recommend to our clients to prepare a roaster of different ciphers, of different strength, where the only requirement from them is that they would be decoy tolerant (as explained before) and generate a random-looking ciphertext (random looking string S). An adversary then will not only not know the payload key, Kp, but also not know the payload ciphertext Cp. To crack the payload S the adversary will have to identify Cp and Kp. This represents a cryptanalytic barrier.

We now assume that within some interval of time of reference ΔT, Alice and Bob exchanged t messages, only s of them were "armed" with a payload. The other (t-s) messages were encrypted with a randomized order of the K1 encrypted letters and the K2 encrypted letters. The adversary will not be able to distinguish between the s payloaded messages and the t-s decoy messages. Consequently, the adversary will try to cryptanalyze all t messages. Failing to extract any message from at least the (t-s) messages, the adversary will not know if Alice and Bob used a payload cipher that resisted their cryptanalysis or that these messages are "innocent".

The big security punch though is projected through the target environment where Alice and Bob are two non-descript communicators in a HIPS message platform where millions of users are sending tens of millions messages daily. All the messages are HIPS formatted; a tiny portion thereto is "payloaded".

Let's say that this HIPS message platform has u users (subscribers). Each user on average exchanges tv messages with v other users over ΔT. This computes to 0.5uvt messages, only a fraction r of them is payloaded. For any standard size message platform this environment totally overwhelms even the most powerful adversary one can imagine. And to the extent that the S string is always random looking, this protocol is not vulnerable to the single smart mathematician syndrome, volumes of randomness safeguard the HIPS users.

One may note that for the millions of users who simply send innocent messages between them, there will be no extra burden. Transmitters may not even know that the text they type is HIPS transferred to their intended recipient, neither does the recipient know in which format the text was carried over from the transmitter. The data at rest at both ends bears no evidence to the privacy preserving secret messages that were exchanged.

Alice and Bob, two random residents of cyberspace will be able to use such HIPS communication platform to exchange secret messages where not the content, and neither the fact that a secret message was passed is exposed against their will.

The more users subscribe to this messaging platform the more the secret message exchangers project security.

Negotiations are in place with message platforms to offer the HIPS advantage.

6.1. HIPS Dedicated Text Messaging Platform

HIPS effectiveness is directly proportional to the volume of use of HIPS communication protocol. It stands to reason then, to dedicatee text messaging like WhatsApp, X, or Telegram, to be operated with a HIPS protocol. The innocent users will see no difference, they type as usual and see messages on the screen as usual. The HIPS distinction will be found under the hood. A HIPS App user will be able to type in a secret code, payload, that would be used to 'arm' the clear text on its move, and will be lost once the original message is decrypted. All data at rest is 'innocent' namely without the payload. The payload will be displayed only on the computing device of the intended reader where the payload secret key (not to be confused with the cleartext keys) is installed. The payload message may remain displayed for a short time. The recipient will be able to copy and pass it over to a safe computing device.

6.2. A Note on Impact

In oppressive societies encryption is not very useful because its users are often coerced to expose its content. The only way to claim freedom in such environment, is to use a means where not only the contents of messaging is obscured, but also it remains hidden that a secret message came to pass [23].

Decoy tolerant ciphers are very common among pattern devoid ciphers (PDC), for example BitFlip [21].

References

1. Samid, G. (2021). AI Assisted Innovation.
2. Samid, G. (2023). Pattern Devoid Cryptography.
3. Mishra, R., & Bhanodiya, P. (2015, March). A review on steganography and cryptography. *In 2015 international conference on advances in computer engineering and applications* (pp. 119-122). IEEE.
4. Taha, M. S., Mohd Rahim, M. S., Lafta, S. A., Hashim, M. M., & Alzuabidi, H. M. (2019, May). Combination of steganography and cryptography: A short survey. *In IOP conference series: materials science and engineering* (Vol. 518, No. 5, p. 052003). IOP Publishing.
5. Challita, K., & Farhat, H. (2011). Combining steganography and cryptography: new directions. *International Journal on New Computer Architectures and Their Applications (IJNCAA), 1*(1), 199-208.
6. Kumar, P., & Sharma, V. K. (2014). Information security based on steganography & cryptography techniques: A review. *International Journal, 4*(10), 246-250.
7. BhanuRajeshNaidu, K., Manikanta, J., Vaseem, S. M. D., Adnan, S. M. D., & Kumar, C. N. (2024). Secure file sharing system using image steganography and cryptography techniques. *In Challenges in Information, Communication and Computing Technology* (pp. 120-124). CRC Press.
8. Abikoye, O. C., Ojo, U. A., Awotunde, J. B., & Ogundokun, R. O. (2020). A safe and secured iris template using steganography and cryptography. *Multimedia Tools and Applications, 79*(31), 23483-23506.
9. Alabdulrazzaq, H., & Alenezi, M. (2022). Performance Analysis and Evaluation of Cryptographic Algorithms: DES, 3DES, Blowfish, Twofish, and Threefish. *International Journal of Communication Networks and Information*

Security (IJCNIS), 14(1).
10. Ammour, B., Bouden, T., & Boubchir, L. (2018). Face–iris multi-modal biometric system using multi-resolution Log-Gabor filter with spectral regression kernel discriminant analysis. *IET biometrics, 7*(5), 482-489.
11. Bagane, P., Venkatesh, S., Guttikonda, J. B., Badhoutiya, A., Pratap Srivastava, A., Khan, A. K., ... & Shrivastava, A. (2024). Securing data in images using cryptography and steganography algorithms. *Int J Intell Syst App Eng, 12*(15s), 17-25.
12. Biu, H. A., Husain, R., & Magaji, A. S. (2018). An enhanced iris recognition and authentication system using energy measure. *Science World Journal, 13*(1), 11-17.
13. Hambali, M. A., Abdulganiyu, O. H., & Lawrence, E. (2022). Enhance Facial Biometric Template Security using Advance Encryption Standard with Least Significant Bit. *Journal of Computer Science and Engineering (JCSE), 3*(2), 60-70.
14. Das, S. B., Mishra, S. K., & Sahu, A. K. (2019). A new modified version of standard RSA cryptography algorithm. In *Smart Computing Paradigms: New Progresses and Challenges: Proceedings of ICACNI 2018, Volume 2* (pp. 281-287). Singapore: Springer Singapore.
15. Islam, M. N., Islam, M. F., & Shahrabi, K. (2015). Robust information security system using steganography, orthogonal code and joint transform correlation. *Optik, 126*(23), 4026-4031.
16. Agu, E. O., Ogar, M. O., & Okwori, A. O. (2019). Formation of an improved RC6 (IRC6) cryptographic algorithm. *International Journal of Advanced Research in Computer Science, 10*(4), 34-38.
17. Jassim, M. F., & Shimal, A. F. (2022). Biometric iris templates security based on secret image sharing and chaotic maps. *International Journal of Electrical & Computer Engineering (2088-8708), 12*(1).
18. Ogundokun, R. O., & Abikoye, O. C. (2021). A safe and secured medical textual information using an improved LSB image steganography. *International Journal of Digital Multimedia Broadcasting, 2021*(1), 8827055.
19. Seth, B., Dalal, S., Jaglan, V., Dahiya, N., Agrawal, A., Sharma, M. M., ... & Verma, K. D. (2021). Secure Cloud Data Storage System Using Hybrid Paillier–Blowfish Algorithm. *Computers, Materials & Continua, 67*(1).
20. Sharma, M. H., MithleshArya, M., & Goyal, M. D. (2013). Secure image hiding algorithm using cryptography and steganography. *IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN, 13*(5), 1-6.
21. Samid, G., & Popov, S. (2017). Bitflip: A randomness-rich cipher. *Cryptology ePrint Archive*.
22. Samid, G. (2023). "Tesla Cryptography:" Powering Up Security with Other Than Mathematical Complexity. *Cryptology ePrint Archive*.
23. NIST Post Quantum Cryptography -- Wrong Headed?.