

Embedding Elliptic Curve Cryptography and Twofish Algorithm to Improve Data Security in Internet of Things

G. Dhamodharan¹; Dr. S. Thaddeus²; Leopoldo, Choque Flores³; Jorge Luis Hilario-Rivas⁴ and Fernando Sandoya⁵.

^{1,2}PG & Research Department of Computer Science,
Don Bosco College (Co-Ed), Guezou Nagar, Yelagiri Hills,
Affiliated to Thiruvalluvar University.
Email: haidhamo@gmail.com

³Universidad Cesar Vallejo, Lima, Perú.

lchoquef@ucv.edu.pe
<https://orcid.org/0000-0003-0914-7159>

⁴Universidad Nacional de Ucayali, Perú.

dr@jorgeluishilario.com
<https://orcid.org/0000-0003-1283-5630>

⁵Instituto de Ciencias Matemáticas Escuela Superior Politécnica del Litoral, Ecuador.

fsandoya@espol.edu.ec
<https://orcid.org/0000-0002-0011-4003>

Abstract

The implicit idea of the IoT is to follow client's identity effectively, so the security and protection concerns like theft the information and disturbance of tasks are turning out to be basic issues in the present IoT applications. Aggressor can pick up the access to the system can harm the physical gadgets of IoT can harm the system which bargains the protection and security of IoT. As the IoT gadgets have low memory, less force and data transmission a productive arrangement to give security is required that won't chomp through the resources of IoT. Because of the resource constrained condition in IoT the regular algorithms isn't sufficient to guarantee the information security. So we want a less computational expense as far as force utilization, memory management and more effective cryptography algorithms to offer more information security in IoT. This paper shows the hybrid cryptography security in IoT by embedding of ECC and TwoFish calculation, which gives an enhanced answer for upgrade the security highlights of the IoT with the goal that we can improve the administration therefore achieves growing the trust over the advancement.

Keywords: Elliptic Curve Cryptography, Encryption, Decryption, Internet of Things, Security, Twofish Algorithm

I. Introduction

Internet of things (IoT) is recognized by free movement of information among various low-power embedded contraptions that use the Internet to talk with one another. It is foreseen that the IoT will

be commonly sent and will find irrelevance in various everyday issues. Solicitations of IoT have as of late pulled in massive thought, and affiliations are amped up for the business estimation of the data that will be made by sending such frameworks. In spite of what may be normal, IoT has distinctive security and insurance stresses for the end customers that limit its development.

Expanding intricacy of IoT arranges likewise amplifies the security challenges looked by such systems. The multifaceted nature of IoT systems is credited to the colossal measure of gadgets associated with the Internet alongside enormous information created by these gadgets. Assaults in IoT are conceivable as the gadgets in the IoT organize are an obvious objective for interruption. Once undermined, the programmers can pick up control and complete malevolent exercises and assault different gadgets near the undermined hub. IoT gadgets don't have infection security or malware assurance programming. This is a characteristic result of the low-memory and low-power nature of these gadgets. The inaccessibility of infection and malware assurance on IoT gadgets makes them profoundly vulnerable to become bots and complete malignant action to different gadgets in the system [7] [8].

When an IoT gadget is hacked, the assailant can likewise seize the steering and sending activities of the gadget. Notwithstanding assaulting different gadgets in the system, aggressors can likewise access touchy information gathered and communicated by the IoT gadgets. This absence of secrecy, uprightness, and security of information in IoT can possibly disturb the far reaching selection of this innovation. It is clear from the conversation till now that the issue of making sure about IoT gadgets is hugely irritated because of their asset obliged nature, because of which answers for assault moderation and security insurance utilized on conventional systems can't be promptly conveyed on IoT systems.

II. Related Work

The impediments of the IOT gadgets are vitality utilization and computational force. Execution of solid security guidance utilizes heaps of gadget power, which isn't prescribed constantly because of the IOT gadgets which are intended to be little where the hitter life can't be expanded. So the regular cryptography calculations which entail high vitality and memory utilization isn't sufficient for the present IOT applications to guarantee information security and confirmation.

There are two kinds of keys are accessible in cryptography: symmetric key and asymmetric key. Symmetric key encryption doesn't require a similar number of CPU cycles as asymmetric key encryption, so you can say it's normally snappier. In this way, with respect to speed, symmetric trumps unequal. The principal hindrance of the symmetric key encryption is that all social affairs included need to exchange the key used to scramble the data before they can unravel it. As there is just one key in the even encryption, this must be known by both sender and beneficiary and this key is adequate to unscramble the mystery message. Conventional Symmetric Algorithms are AES, DES, IDEA, Triple DES, Blowfish which cannot be utilized to for IOT gadgets because of their bigger key sizes, bigger block lengths and helpless against a few attacks like savage power attack [5].

Asymmetric encryption utilizes lengthier keys than symmetric encryption so as to give preferable security over symmetric key encryption. While the more drawn out key length in itself isn't so much a

hindrance, it adds to more slow encryption speed. In uneven or public key, cryptography there is no obligation for trading keys, subsequently taking out the key dispersion issue. The essential favourable position of public key cryptography is expanded security: the private keys never should be sent or uncovered to anybody. To accomplish the verification the sender encodes the message by his own private key and the information is decoded by the sender's public key at the collector side. The usually used asymmetric algorithms are ECC, Deffie Helmen, Hash Functions and RSA. The constraint of these algorithms is bigger key size which builds the multifaceted nature of the algorithm and expands the computational cost which again isn't practical for the IOT gadgets which cannot manage the cost of for more computational force and vitality utilization [6].

III. Elliptic Curve cryptography

Elliptic Curve cryptography is an open key encryption technique reliant on the elliptic bend theory that can be used to make snappier, more diminutive, and more profitable cryptographic keys. ECC outfits a more raised degree of security with lesser key size appeared differently in relation to other Cryptographic methodology [5]. It is characterized by the scientific condition

$$Y^2 = x^3 + ax + b$$

where $4a^3 + 27b^2 \neq 0$

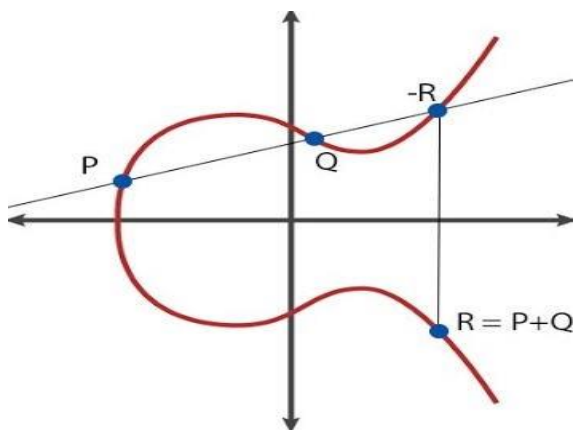


Fig 1: Elliptic Curve $Y^2 = x^3 + ax + b$

ECC-Based Secret Key Derivation using ECDH

Encryption Key Calculation

Step1: Generate ciphertextprivatekey = new random private key.

Step2: Compute ciphertextpublickey = ciphertextprivatekey * G.

Step3: Compute the shared secret: sharedECCkey = publickey * ciphertextprivatekey.

Step4: Return both the sharedECCkey + ciphertextpublickey.

Decryption Key Calculation

Step1: Compute the shared secret: sharedECCkey = ciphertextpublickey * privatekey.

Step2: Return the sharedECCkey and use it for the decryption.

IV. Twofish

Twofish is a symmetric key square code with 128 pieces and key sizes up to 256 pieces. It utilizes pre-figured, key-subordinate S-boxes. Twofish is quick on both 32-piece and 8-piece CPUs and in gear. Furthermore, it's adaptable; it will all in all be utilized in sort out applications where keys are changed a noteworthy piece of the time and in applications where there is in every way that really matters no RAM and ROM accessible. There are 3 stages in Twofish calculation, the hidden development is discrete information bit into 4 sections, the subsequent development was performed XOR activity between bit responsibility with a key, and the third step setting up the information bits in numerous occasions Feistel sort out. Beginning at now there is no gainful cryptanalysis of Twofish [4,1].

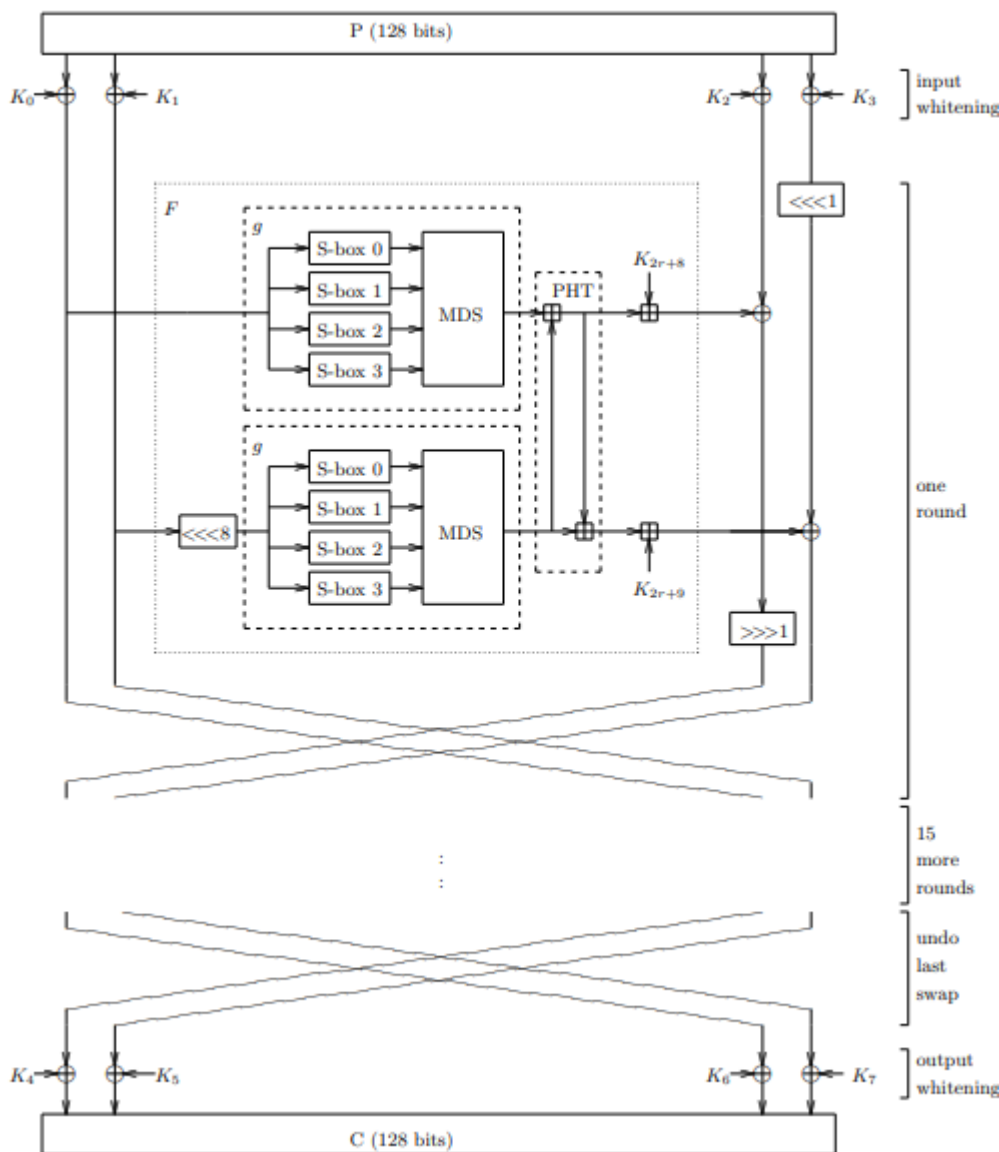


Fig 2: Twofish

In each round of Twofish, two 32-piece words fill in as promise to the F fuction. Each word is confined into four bytes. Those four bytes are sent through four varying key-subordinate S-boxes. The four yield bytes are joined utilizing a Maximum Distance Separable (MDS) mastermind and consolidated into a 32-piece word. By then the two 32-piece words are joined utilizing a Pseudo-Hadamard Transform (PHT), added to two round subkeys, by then XORed with the correct portion of the substance. There are also two 1-piece unrests going on, one proceeding and one after the XOR.Steps involved in Twofish calculation are

Step 1: Bit commitment as much as 128-bit would be parcelled into four fragments, each for 32 bits using little-endian show. Two bits of the bit will be the right part; the two bits of various pieces will be left.

Step 2: Bit-XOR commitment to advance with the four key parts. $R0,j=Q \oplus K_j$; $j=0,1,2,3$ Where K is the key, K_j implies the sub key where $i=0,1,2,3$.

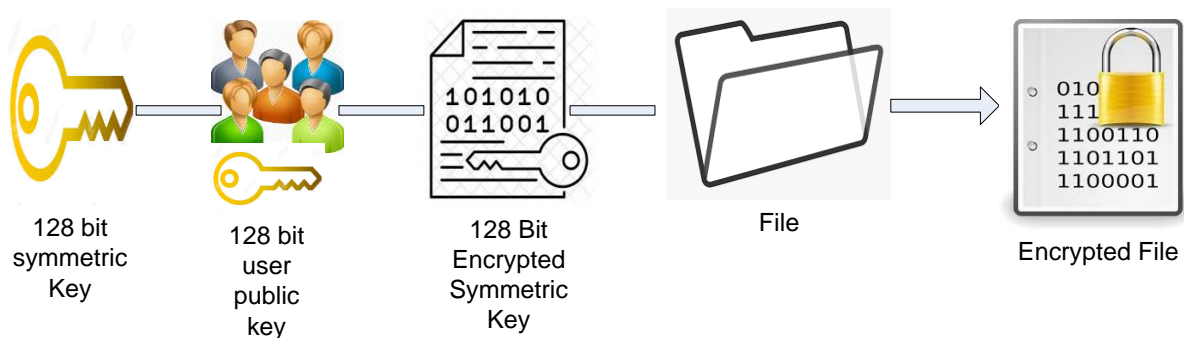
Step 3: Twofish calculation utilizes a Feistel network structure. Feistel system utilized by Twofish comprises of 16 cycles. Function f of Twofish comprises of a few phases:

- i. Function g, which comprises of four s-box and MDS matrix
- ii. IPM (Pseudo Hadamard Transform)
- iii. The expansion of the key aftereffects of IPM

V. Proposed System

Half breed Cryptography encryption is a joined strategy of two Encryption Algorithm so as to offer security to information in IoT, In this paper Symmetric and Asymmetric sort encryption framework are acknowledged to get a valuable outcome. The proposed strategy as appeared in Figure 3 works so it orchestrates the speed of one key encryption and interpreting in relationship with the security that both Public and Private Key gives, which along these lines understands a well secure sort of encryption. Half breed Cryptography performs by encoding the information with a Symmetric Key which will be then blended in with an Asymmetric Key of the sender. To decipher the mixed information, the recipient should from the start interpret the open key with the gave Asymmetric Key and a brief timeframe later utilize the Public Key to disentangle the information which is been gotten [2].

Hybrid Encryption



Hybrid Decryption



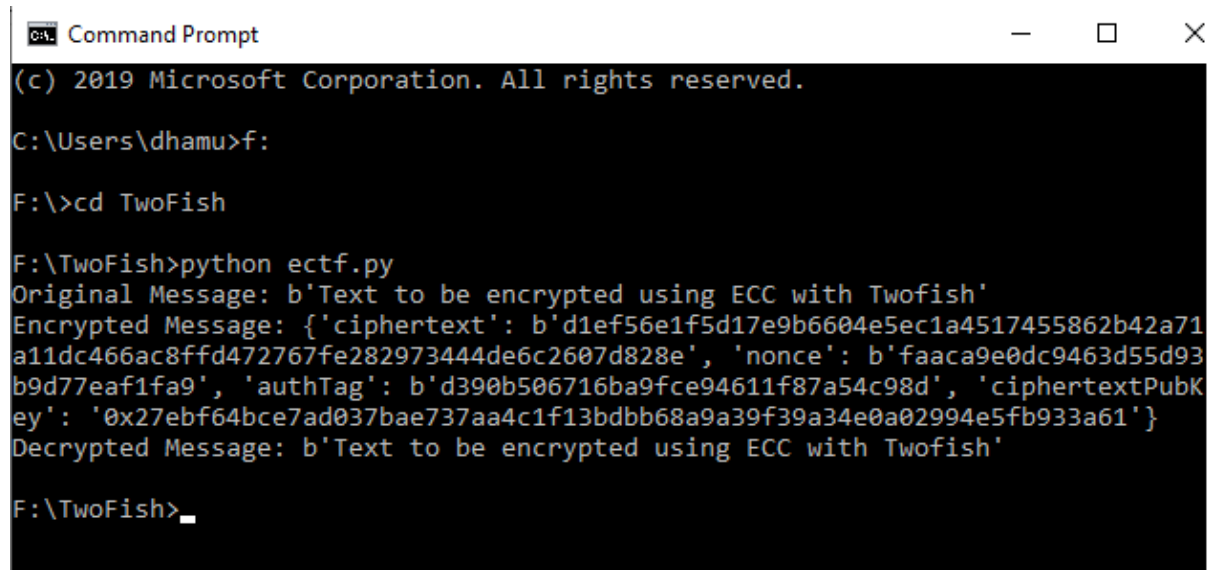
Fig 3: Hybrid Encryption/Decryption

Symmetric encryption of Twofish is utilized to change over the plaintext to encode text. This exploits the symmetric encryption speed. Asymmetric encryption is utilized to trade the symmetric key utilized for encryption. This exploits the security of awry encryption, guaranteeing that solitary the planned beneficiary can decode the symmetric key using ECC.

Steps involved in Elliptic Curve Twofish for Encryption/Decryption:

- Step 1: The sender recovers the recipient's public key using ECC.
- Step 2: The sender generates a symmetric key with Twofish
- Step 3: The symmetric key is encrypted with the public key of sender using ECC
- Step 4: Encrypt the plaintext with encrypted symmetric key using Twofish
- Step 5: Decrypt symmetric key with ECC's private key of receiver
- Step 6: Decrypt cipher text with twofish's symmetric key

Implementation of ECC with Twofish



```
Command Prompt
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\dhamu>f:
F:\>cd TwoFish
F:\TwoFish>python ectf.py
Original Message: b'Text to be encrypted using ECC with Twofish'
Encrypted Message: {'ciphertext': b'd1ef56e1f5d17e9b6604e5ec1a4517455862b42a71a11dc466ac8ffd472767fe282973444de6c2607d828e', 'nonce': b'faaca9e0dc9463d55d93b9d77eaf1fa9', 'authTag': b'd390b506716ba9fce94611f87a54c98d', 'ciphertextPubKey': '0x27ebf64bce7ad037bae737aa4c1f13bdbb68a9a39f39a34e0a02994e5fb933a61'}
Decrypted Message: b'Text to be encrypted using ECC with Twofish'
F:\TwoFish>
```

Each development of the round limit is bijective. That is, each yield is conceivable. We've seen an over the top number of assaults against figures that don't have this property not to combine it. The round capacity stirs up assignments from various logarithmic social events: S-box replacement, a MDS structure in GF(28), improvement in GF(232), augmentation in GF(2), and 1-piece turns. This makes the estimation hard to snare deductively. Key-subordinate S-boxes were not picked discretionarily. Or on the other hand perhaps, we intentionally sorted out S-box improvement controls, and endeavored them with all conceivable 128-piece keys to ensure that all the S-boxes were in actuality solid. This framework permitted us to join the idea of fixed, solid S-boxes with the idea of question S-boxes. Plus, Twofish has no delicate keys. No other computation has a practically identical adaptability in execution: the capacity to deal key-game plan time for encryption speed and ROM and RAM for encryption speed and it improves the level of security with ECC.

VI. Conclusion

In future IoT turns into the most fundamental piece of the humanity for quality life. Huge measure of delicate information is conveyed between the gadgets with asset limitations like less memory space, low force where the information security is the fundamental concern. The customary calculations which requires more computational force and memory which isn't adequate for the current situation with IoT. Henceforth light weight cryptography calculations are need. It is basic to build up a more made sure about light weight cryptography calculations that has low computational cost, all the more preparing speed and littler key size. This Proposed work infers that the arrangement gave by the crossover encryption utilizing ECC and Twofish gives a superior arrangement when contrasted with other encryption calculations. This solution can be applied in IoT to give a superior security to the information.

References

1. Acharya, Kritika, Manisha Sajwan, and Sanjay Bhargava. "Analysis of Cryptographic Algorithms for Network Security." *International Journal of Computer Applications Technology and Research* 3, no. 2 (2013): 130-135.
2. Eisenbarth, Thomas, Sandeep Kumar, Christof Paar, Axel Poschmann, and Leif Uhsadel. "A survey of lightweight-cryptography implementations." *IEEE Design & Test of Computers* 24, no. 6 (2007): 522-533.
3. MUCH AZIZ MUSLIM, ILKOM UNNES, ILKOM UNNES BUDI PRASETIYO, and ILKOM UNNES ALAMSYAH. "Implementation twofish algorithm for data security in a communication network using library chilkat encryption activex." *Journal of Theoretical and Applied Information Technology* 84, no. 3 (2016): 370-375.
4. Siva, Sankaran P, and V. B. Kirubanand. "Hybrid cryptography security in public cloud using TwoFish and ECC algorithm." *International Journal of Electrical and Computer Engineering* 9, no. 4 (2019): 2578.
5. Singh, Laiphrakpam Dolendro, and KhumanthemMangle Singh. "Implementation of text encryption using elliptic curve cryptography." *Procedia Computer Science* 54 (2015): 73-82.
6. Schneier, Bruce, John Kelsey, Doug Whiting, sDavid Wagner, Chris Hall, and Niels Ferguson. "Two sh: A 128-bit block cipher." AES submission (1998).
7. Goyal, Tarun Kumar, and VineetSahula. "Lightweight security algorithm for low power IoT devices." In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1725-1729. IEEE, 2016.
8. Tewari, Aakanksha, and B. B. Gupta. "A lightweight mutual authentication protocol based on elliptic curve cryptography for IoT devices." *International Journal of Advanced Intelligence Paradigms* 9, no. 2-3 (2017): 111-121.