
Proposed Hybrid RSA, AES and SHA Algorithms with Hardware Key Security Mechanisms for Embedded Systems with Internet of Things**J.S.Prasath¹**jsprasath@gmail.com¹¹Department of Computer Science and Engineering, Sapthagiri NPS University, Bengaluru, India

Article Information

Received : 1 Jun 2026

Revised : 22 Jun 2026

Accepted : 30 Jun 2026

KeywordsInternet of Things,
hardware, software,
security, embedded
systems

Abstract

Embedded systems are fast growing technologies and it is widely used in varieties of applications. It is essential to protect the embedded system along with wireless networks from unauthorized access and modification of process parameters. The novelty of this proposed work is the implementation of hybrid security algorithm along with the exclusive hardware key using embedded system. This proposed hybrid security algorithm is a mixture of asymmetric, symmetric and hash function algorithms which assures strong security. The process data security is strengthened by generating large key size of 2048-bit in this proposed asymmetric encryption and 256-bit key size for symmetric encryption. The hardware key is essential to access and modify the encryption code as well as to monitor the cipher text both at the transmitter and the receiver. This proposed security algorithm is cost effective and it prevents the sensitive process information from illegal access and alteration.

A. Introduction

Embedded systems are used in wide range of products and are employed in various applications. Embedded devices often need to gather the information, to store the data and communicate the sensitive information to the outside world. The security architecture of embedded system is subjected to numerous resource constraints which includes memory, computational capability and power. The security threats are increases due to the usage of embedded systems in a wide range of applications. The security of embedded systems can be performed at several stages which include protocol, algorithm, architecture and circuit. The information communicated between process and embedded devices may be hacked or altered by the intruders. The integrity of information results not only damages the physical components but also malfunctioning of entire plant operations. The implementation of security in hardware components, software coding and communication is a major concern for embedded system. The security of embedded system is assessed and enhanced with respect to various product development phases [2]. The security is accomplished through the analysis of hardware attacks, security requirements, and provides solutions for protecting the embedded memories. The communication security is attained with on-chip and design of system-on-chip is managed. It is essential to propose the security mechanism for embedded hardware and to prevent the modification of information from unauthorized access.

Information and communication security of embedded system has gained significant attention. The architecture of embedded system should be flexible to support the security functionalities and standards. The implementation of security mechanism in embedded system becomes complex due to its limited processing power and memory capacity. The energy-efficient security protocols are essential in order to maximize the processing capability of the embedded system. The various attacks on embedded systems are energy drainage, physical intrusion, network intrusion, information theft, damaging of sensors or peripherals and systems reprogramming. Hardware based attacks such as physical attacks, system-on-chip and networks-on-chip attacks cause serious problems and results in failure of embedded systems. The deviation of program behavior due to attack is monitored by the hardware mechanism and it operates parallel to an embedded processor [14]. The hardware monitor verifies the operation steps which are performed that match up with the installed application. The hardware alerts the user whenever attack disturbs the execution step pattern. The hardware based security solutions need to be proposed to overcome the limitations of embedded architecture.

The hardware based side channel attack monitors the physical behavior of the embedded system. Software based attacks such as code injection attacks and cryptographic attacks disturb the normal functioning of plant operations. A cost effective method is proposed for protecting embedded software against passive side channel attacks [7]. The performance is validated using ARM Cortex-M4 for standard block ciphers. The protocols are vulnerable to software attacks and could result in a malevolent behavior such as unknown destination, packet replay or deadlock. The buffer overflow attack becomes dangerous to application-specific embedded systems. The insecure code in buffer overflow attack stores inputs into

a buffer and allows overwriting data on the stack by the attacker. The software based side-channel attacks monitors the logical behavior of the embedded system. These attacks extract the confidential information and monitors the data traffic or to facilitate the observation process by adding a malicious code. Embedded systems are designed to perform any specific pre-defined task that must meet any real time constraints. The information in a network passes through a number of untrusted connection. During the design phase of the embedded system, security is not considered and it is complex to implement once the manufacturing process is finished. The information security requires during data transmission and accessing at the end user devices. It is essential to implement the protocols for secure data transmission in embedded systems. The security mechanism should be implemented to ensure data security against unauthorized access. The incursion on embedded system gives rise to failure of physical components, equipment damage, loss of revenue and unsafe working environment. The designer of embedded system should integrate security mechanism with embedded devices. The risk evaluation is essential to recognize the security hazards and vulnerabilities in the embedded system. The attacks on embedded system are split into hardware and software. The physical intrusion takes place with the use of probe to observe inter-component communications and reverse engineering of a chip or a printed circuit board (PCB). The software incursion occurs due to the coding attacks. A strategy proposed for run-time monitoring of code integrity in application-specific processors [14]. The execution trace is monitored at run-time and examines whether it synchronizes with the anticipated program behavior. This proposed system can be used to discover the program code integrity over a wide range.

Embedded systems are coupled to the open networks such as internet due to the advancement of Information and Communication Technologies (ICT). This builds new security issues for embedded systems. It raises the security threat and cyber-attacks on embedded systems. The risk assessment and risk management is essential in order to safe and secure usage of embedded systems. A Dynamic Security Risk Management (DSRM) mechanism is proposed which are used for networked industrial applications to manage the intermittent real-time process [5]. The security of real-time embedded systems is upgraded with respect to the industrial networks. It is essential to model the embedded system with assorted security and integrity protections.

The intrusion that takes place on the normal control flow of the programs running on the embedded device called as control hijacking attacks. Attackers try to inject the code which is executed on the embedded system. The various categories of physical attacks are presented on security related embedded systems [9]. The physical attacks are categorized into Invasive, Semi-invasive and Non-invasive attacks. The aim of these attack is to physically disturb the embedded chip operation, observing the behavior of embedded chip and to extract about the processed data such as power dissipation, computation time etc. It is essential to enhance the security of embedded system by considering these attacks. The attacker can also access the address and data lines, can observe and override bus transactions to perform splicing, spoofing and replay attacks.

Embedded systems are widely used in varieties of applications from safety-critical systems to data collection in harsh environments. Embedded systems are vulnerable to varieties of attacks which include physical intrusion, network intrusion, energy drainage, information theft etc. The security attacks take place at design level, implementation level and testing level. A self-organizing approach is proposed which detect the abnormal behavior of program [6]. The algorithm is developed to identify the program behavior and to increase the security level of embedded systems. The system of attack may also vary for users, content providers and manufacturers. It is necessary to design and implement the security aware embedded system. Embedded systems exchange data and sensitive plant information locally as well as globally using networking and wireless communication. It is mainly used to capture, store, process and transmit data in various applications such as automobile, trains, consumer electronics, medical equipment, network appliances, smart cards, mobile phones etc. The usage of internet and wireless technologies in embedded system applications increases the varieties of attacks. The cryptographic algorithm is essential to secure the critical plant information which is communicated using embedded system with wireless networks. A multi-dimensional analysis framework is proposed for cryptographic algorithms in terms of speed, power, and unit energy cost [10]. The experimental result and analysis predicts that the plain text size is not linear with the energy consumptions and time overheads of cryptographic algorithms. The security algorithm characteristics are analyzed based on experimental data from an embedded platform. Data security plays a crucial role in the design of embedded systems. Intruders can also gain access to the memory contents from an external programmable read-only memory (PROM) or external RAM memory. The data stored in memory should be encrypted and authenticated. The contents of memory can be modified by the attackers. The energy efficient approach called Time Stamps Verification (TSV) is proposed to perform Memory Integrity Verification in embedded systems [11]. The memory integrity attacks are splicing attack, spoofing attack and replay attack. The hash function is used to protect the data integrity. The message authentication code should be used to solve the memory integrity problem. The pseudo-random number generator is used to generate the timestamps. The security mechanism is essential for embedded system with energy efficiency should be taken into consideration.

The challenges in securing the embedded systems arise from developing the architecture till implementation. The security requirements of embedded system are data confidentiality, integrity, authentication, privacy, denial of service, hardware, software etc. The security requirements, attack techniques and its countermeasures for embedded systems are proposed [8]. Physical attacks can be launched by probe to eavesdrop on communications in circuit board of embedded systems. Micro-probing techniques can be used for launching attack on system-on-chip. The confidentiality and integrity of information and code is to be ensured against software attacks. The required countermeasures are accesses regulation of various software components such as operating system, downloaded code, etc. to different portions of the system including registers, memories, co-processors etc. It is essential to develop the secure platform for embedded system to ensure security at both hardware level and software level. The countermeasure technique is

proposed which are based on fault space transformation to protect AES-128 bits against biased fault attacks [1]. The fault collision based attacks are prevented. The time and hardware redundant versions of AES (Advanced Encryption Standard) are performed for biased fault attacks. The fault is introduced in the number of rounds in AES encryption and exploits the biased nature of the introduced fault to decipher the key. The information which is communicated across embedded system with wireless networks should be confidential and it is accessed only by the authorized users. Confidentiality is achieved by encrypting the process data using suitable encryption algorithm. The confidentiality and integrity of process data is essential across embedded networks. Data integrity algorithms are necessary to check for any alterations in the transmitted information. Integrity algorithms are used to keep data privacy, to check for any changes in process information and to make data accessible to authenticated users. A cost-effective, flexible architecture is proposed for medium to high level embedded processors that ensures integrity and confidentiality [15]. The configured processor operates in a secure mode which permits execution of only trusted programs i.e. Code integrity. It is essential to propose the high performance hardware system to protect the data.

Embedded systems are vulnerable to attacks that target data during runtime of the code. Attackers may divert the execution flow of code which is running in the embedded devices. It is essential to monitor the control flow of the program for any variations in the intermediate data and the security mechanism is essential to ensure secure code execution during runtime. A scalable, application-specific methodology is proposed for securing the program execution on embedded processors [16]. The performance of program execution time is monitored by the hardware-assisted architecture. The dedicated hardware monitoring is employed to detect and prevent fortuitous program behavior. The consistency between stable and unstable aspects of a program needs to be considered when designing the embedded hardware.

B. Research Method

This proposed hybrid security algorithm combines the asymmetric, symmetric and hash function cryptography along with the dedicated hardware key which ensures higher level of secure data transmission. It encrypts the hardware key along with the hybrid cryptography in order to strengthen the security. The asymmetric encryption is used in this work which generates 2048-bit random private key. This private key is used as input for symmetric encryption. The 256-bit key size is used for symmetric encryption to generate cipher text.

The modified AES encryption and decryption is performed with increased number of rounds which strengthen the security [3]. The initial key is generated from the Polybius square. The alphabets and numerals are included in the Polybius Square filled without reoccurrence. This security algorithm takes more time for computation and it is hard for attackers to break the system. The asymmetric cipher is generated using RSA (Rivest-Shamir-Adleman) encryption and the symmetric cipher is generated using Advanced Encryption Standard (AES) encryption. The proposed hybrid algorithm generates a random key from the RSA algorithm and a plain text is given as input to the AES encryption to generate

cipher text. The AES encryption is performed for ten rounds and the encrypted data is taken at the end of the tenth round. The cipher text is transmitted across wireless networks using raspberry pi at the transmitter. The key management is achieved by generating the hash value for a given private key using hash algorithm. This hash value ensures data integrity during transmission over wireless networks.

A public-key broadcast encryption scheme is proposed for managing the information sharing between the users of online social networks [12]. The privacy of user's data and their online social relationships can be protected from third parties by an efficient privacy protection framework for Online Social Networks. The security is performed by protecting the confidential information from attackers and access control rights provided to the authorized users. The Facebook based parameters are used for performance evaluation of the proposed framework. The encrypted data is given as input to the two nodes at the receiver. The cipher text is applied as input to AES decryption. The SHA-512 hash algorithm is proposed in this work in order to generate hash value to ensure data integrity. The encryption and decryption of plant information is securely monitored through the internet. This proposed security algorithm can be used to secure sensitive plant information for embedded system with process monitoring through internet. The flowchart describes the flow of encryption and decryption. Figure 1 shows the sequence of operations performed during the process of encryption. The input plain text is first encrypted using asymmetric algorithm with 2048-bit key size to produce cipher text. The output of asymmetric encryption is further given to symmetric encryption with 256-bit key size to generate new cipher text. The dedicated programmable hardware key is utilized that enables transmission of cipher text over internet. The SHA-512 hash algorithm is used to generate hash value in order to ensure integrity of the process information.

Figure 2 shows the sequence of operations performed during the process of decryption. Decryption is the reverse process of encryption. The cipher text obtained at the receiver is converted back into original plain text through the process of symmetric and asymmetric decryption. The efficiency and security of the Supervisory Control and Data Acquisition (SCADA) communication is increased by the new cryptographic key management scheme [13]. The scheme uses two key update phases which are session and master key updates. The master station generates the session keys in the session key update phase. The Elliptic Curve Diffie-Hellman (ECDH) protocol is used in the master key update phase. The modeling of Security Index (SI) and Quality of Service (QoS) is achieved through poisson process. The benefit of this proposed hybrid encryption algorithm is that it generates different cipher text for the same messages by using different initialization vectors. The generation of random key size is larger that is, 2048-bit and this key size can be expanded.

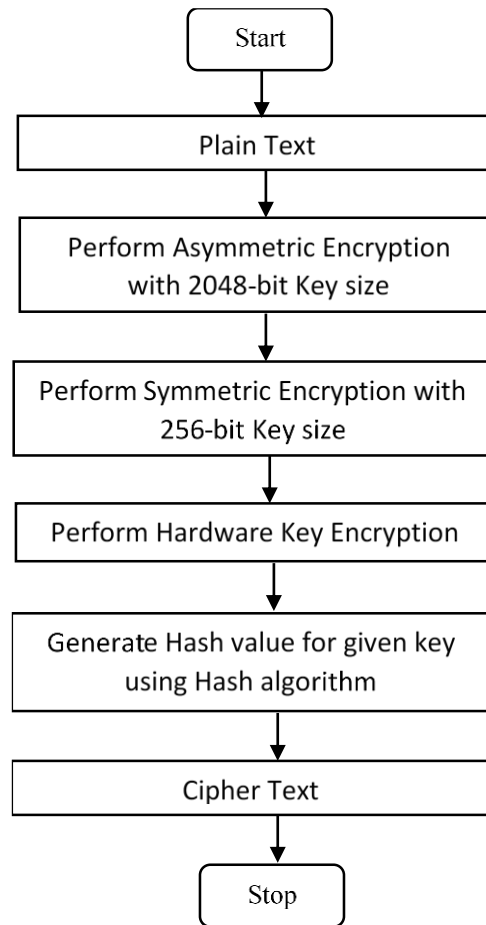


Figure1. Encryption Process

A new cryptographic algorithm is proposed which combine Elliptic Curve Cryptography (ECC) and Advanced Encryption Standard (AES) to provide encryption, XOR-DUAL RSA algorithm for authentication and Message Digest (MD-5) for integrity [8]. The performance of this algorithm is better in terms of processing time, cipher text size and the energy consumption in Wireless Sensor Networks (WSN). It decreases the processing overhead and achieves lower energy consumption. The cipher text is given as input to decryption process. The hardware key is to be decoded and the symmetric decryption is performed with 256-bit key size. After the symmetric decryption, asymmetric decryption is to be performed with 2048-bit key size to obtain the original plain text.

This proposed hybrid cryptographic algorithm combines the features of symmetric, asymmetric, hash function cryptography and the dedicated hardware key. The key size of 2048-bit is used for asymmetric algorithm which is very larger as compared to existing hybrid algorithms. The generated cipher text from this proposed hybrid security algorithm is transmitted only by the authorized user having dedicated hardware key over internet. The cipher text and the original plain text can be monitored through the internet by providing the IP address. This proposed hybrid security algorithm is also implemented and tested in real-time using embedded hardware. The data transmission security in WSN is achieved through novel encryption scheme which is based on Elliptic Curve Cryptography

(ECC) and homomorphic encryption [4]. The optimum network structure is built in the form of clusters using genetic algorithm. The combination of ECC key, node identification number, and distance to its cluster head produces the encryption key and its size is 176-bit. The cluster head is allowed to aggregate the encrypted data using homomorphic encryption without having to decrypt them. The network performance is enhanced in terms of lifetime, communication overhead, memory requirements, and energy consumption.

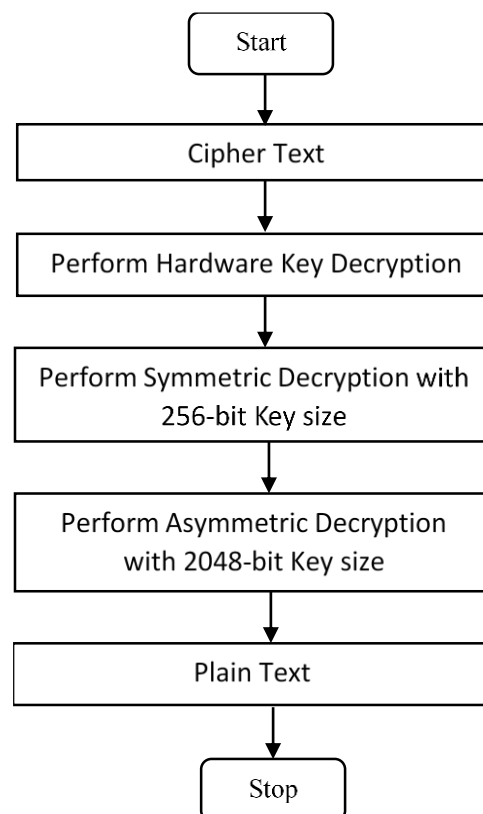


Figure 2. Decryption Process

Figure 3 shows the block diagram of secure monitoring of sensor data using embedded system through the internet. The three nodes are used in this proposed work in which the transmitter node acts as a master and the two receiver nodes acts as a slave. The temperature and gas process data are detected and encrypted at the master node. The hardware key, asymmetric, symmetric and hash algorithms are used to generate the cipher text. The generated cipher text is transmitted across the internet. The two nodes are used at the receiver which receives the cipher text. The decryption algorithm is executed at the receiver nodes to obtain the process data in plain text. When the master node fails, then one of the slave nodes acts as a master and transmits the process data to another slave node. The slave node performs decryption as well as acts as redundant node. The decrypted data can be monitored through the internet by providing the correct IP address.

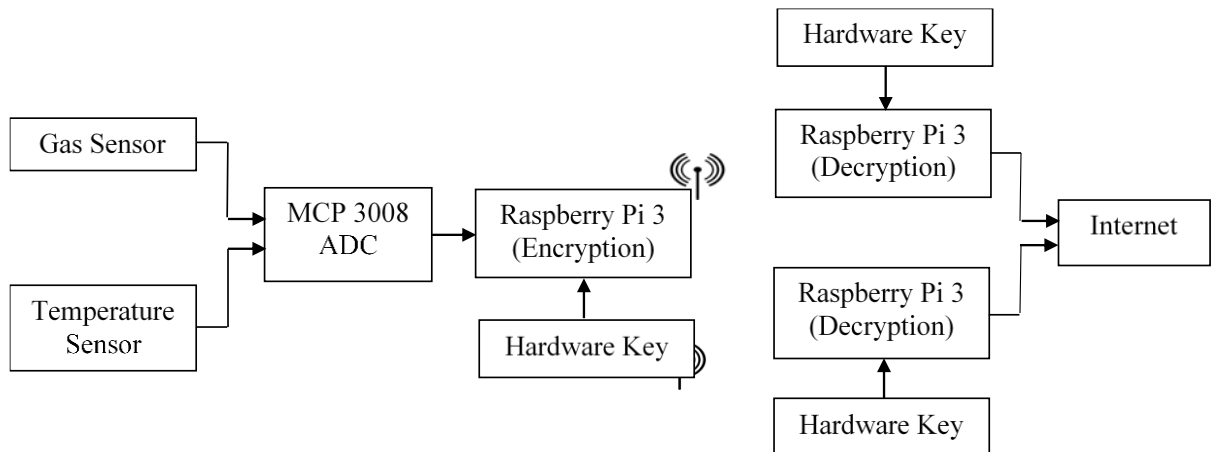


Figure 3. Block Diagram of Secure Data transmission and reception using Raspberry Pi

The temperature sensor LM35 is used which generates an output voltage proportional to the centigrade temperature. The temperature range of LM35 is from -55°C to 150°C . The MCP3008 is a low cost 8-channel 10-bit analog to digital converter. The MCP3008 Analog to Digital Converter (ADC) is connected to the Raspberry Pi using a Serial Peripheral Interface (SPI) connection. The MQ2 gas sensor is sensitive to smoke and detects the presence of flammable gases such as methane, propane, butane, alcohol, hydrogen and liquefied petroleum gas. The MQ2 sensor has four pins which are analog pins, digital pins, power supply, and ground. The raspberry Pi 3 Model B is a low-cost, quad-core small sized single board computer based on ARM cortex-A series processor, widely used for industrial applications. The raspberry Pi 3 includes processor, memory, and connectors for Ethernet, audio output, video output, General Purpose I/O (GPIO) connector, camera, display and slot for memory card. It has inbuilt Bluetooth and Wi-Fi which enables wireless communications. In this proposed work, three nodes are used out of which one at the transmitter side and the other two at the receiver side. The transmitter node performs data encryption and the decryption is performed through two nodes used at the receiver.

C. Result and Discussion

This proposed work is used to read, transmit and receive the sensor data with security using three raspberry Pi. Figure 4 shows the hardware implementation of secure transmission of gas and temperature data using raspberry pi through Wi-Fi. The raspberry pi used at the transmitter reads the gas and temperature sensor data. The hardware key is essential to execute the hybrid encryption algorithm and to generate the cipher text. The hybrid encryption is performed at the transmitter node which includes asymmetric, symmetric and hash algorithms. It also encrypts the hardware key along with the hybrid algorithm to enhance the level of security. The key size of the proposed asymmetric encryption is 2048-bit which are further used for symmetric encryption algorithm for the new key generation. The key size of the proposed symmetric encryption is 256-bit is used to generate the strong cipher text. The SHA-512 hash algorithm is used in this work to ensure data

integrity during transmission over the internet. The IP address is essential to monitor the encrypted data through the internet. This encrypted data is transmitted to the raspberry pi at the receiver through the internet.



Figure 4. Hardware Implementation of Secure transmission of Process data using Master Node through Internet

Figure 5 shows the hardware implementation of secure reception of sensor data at the slave node1. The hardware key is necessary to execute the hybrid decryption algorithm and to monitor the process data in original numerical form. The IP address is essential to monitor the sensor data through the internet.



Figure 5. Hardware Implementation of Secure reception of Process data at Slave Node through Internet

When the master node fails due to some fault, then the slave node1 acts as master and it transmits the encrypted data to the slave node2. The slave node1 acts as redundant node as well as it performs hybrid decryption. This proposed hybrid security algorithm generates asymmetric key of 2048-bit length. The symmetric key is generated which is 256-bit length. The hash algorithm operates on 1024-bit message block size and a 512-bit intermediate hash value. The hardware key is essential to view and run the encryption code. The asymmetric encryption is widely used to protect the sensitive plant information. The two keys are generated in asymmetric encryption. The public key is essential for encrypting the process data to obtain the cipher text and the private key is required for decrypting the process data to obtain the plain text. The strength of the security depends on the key size. In this proposed work, the key size of 2048-bit is

generated in asymmetric encryption which increases the level of security. The unique hardware key is to be inserted in the USB to access and run the encryption file. The inclusion of hardware key with the hybrid encryption greatly strengthens the security.

Generation of Private Key and Public Key

The asymmetric algorithm generates key size of 2048-bit which is given below.

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAvu82MTAjp5HRJ6xLkG8F30/i/1167PSuQ9WuLl/NtIfuj6b/
x0MUbj88dDzLZUtZ09UKnP1ZSJcinApbi5J4icz4t++PkbGE8ryL1dpQ1q0OzOw
mzYV4IecoobtiT1B32b80Jw9+p2vPzgMXyjdR/5WDeFvQejchwE0qmsirepFP1Y
RUcxMWU+JYNWA5Y8Kz9Mjbqf9IrsOwrXuiNikVnPUqMs0JToOlXkhqR+65SQr1w+
H5WbISXUJOcPK1AMwncyqVoc6ZyQnLuTAW3j9l+zi/Isf1hhfA+Xvf6noGfxWifb
Nda/lm/GiuRpA3t42OeV+t49B8kV4QjUf7p+SwIDAQABAoIBACLyL24DR9EWvDxR
qabu2xjcAySMVJJoGjkqM+4i9Mh1FLHQNAJpgaTICC2SLvwENZWy6XVgXWYNquV6
c9Tw4wT9rPQNVYmadliM3ZTSn96HBX9Kb6mI1ao5x+5rWbCDuLnF5KYL2sAy+i9M
995qvW1+3ls/1Tfg3QZ2oF9aVM0yJNJZrCUxgpX7QnSJ/q3p/LHjclaeJ68sCWxP
0skOR34upPoCQFDR2iTkVItT6SdiFg5uYc4sIjPkak1Lvxd809C55TRmqgFzwTGs
6WgrrfMNZaJs7mldyiNb/7q0YAkmlEldhLx6zJ4FSjyYfnTfiK6Ud11wOuJ303V3
RjREHlkCgYEAxM+P2wCfjH+WKVyFSNbDeNlbbcWkl33LLQrrzZoUt0gWLFWkCBli
eYzXxriV725FoDwnx9BKz8hiu1GQVtkV/2XZNinEN9nenrIoqUTg1w7LAjEG+lm3
gMzVcNxfLzKcNukeApym1LqYxpYA6ZzWv6cP6Yx3peqXmtGgw9km3UCgYEA+Fs6
mVCSOG+xqYt0AVp7PzFckGlvdmi4LRLh2ov7cvg6/NyEhG8o7AdUh/jTpED0o
NOFDKA6Kv9Wp/TV+QWqUxJfQ9yh6Kp7HVySuZGmTotf9nJJEKmk//WeTVleeGkY
LLi7XZmis8zsyXjRye7CIRV81TbFW7zQIjJqOr8CgYEAwpCDobjzAf2b4JXzufd+
1PjkNxdyBn3Azd5uA9sDWk5amjcIGL9K/b91hnjOAlm0arGLPWpD9MFMDPJluKH
qa3xV2dg8On0+CQeqIBMwq7oKOI5AYQyMiXNmD5z5v/8iD1oQbfqZv1lc+K2Qh2K
itmMdCwDvw9mwOMCmUxdZIUcGyEA63ftfzjc6ZU8pLsLKZ1X7Pon/tgzhWw1VWCa
Y46MLM3nDGt+8A4BDhAXEg25RQN1VWgmjsKOTZQ8CNXTbbciFXOvyRXvlJqDB81b
HOOAWWC2hRGv0kFY9OU5V1TQ2hU5dIBv4zbirkPT9xT8C6j5frNYa3Wr/ZsIU4oY
69NQ1JECgYBA89+4ycSo3gWkVC05z7Vor0f+nRpt8UNAEtuM6jpK+58e+qS2Njlp
79HdupMY+C7rF3pjRK9t4Jq4EYjxtjvkDKt1zzZYOf6Bv8fyKzgibLd0l+1DIIUn
miFCDgkENvicb/d4LaGVXQwAXXKncAyDut83hAeaukmjZ372kbqpZg==
-----END RSA PRIVATE KEY-----
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvu82MTAjp5HRJ6xLkG8F
30/i/1167PSuQ9WuLl/NtIfuj6b/x0MUbj88dDzLZUtZ09UKnP1ZSJcinApbi5J4
icz4t++PkbGE8ryL1dpQ1q0OzOwmzYV4IecoobtiT1B32b80Jw9+p2vPzgMXyjd
R/5WDeFvQejchwE0qmsirepFP1YRUcxMWU+JYNWA5Y8Kz9Mjbqf9IrsOwrXuiNi
kVnPUqMs0JToOlXkhqR+65SQr1w+H5WbISXUJOcPK1AMwncyqVoc6ZyQnLuTAW3j
9l+zi/Isf1hhfA+Xvf6noGfxWifbNda/lm/GiuRpA3t42OeV+t49B8kV4QjUf7p+SwIDAQAB
-----END PUBLIC KEY-----
```

The symmetric encryption is used in this work which is based on the number of transformations to be performed on the stored array. The AES-256-bit key size is used in this work which performs 14 rounds of encryption. The first transformation is the data substitution using a substitution table. The second and third transformations are shift rows and mixes columns respectively. The final transformation is to perform Exclusive-OR (EX-OR) operation on each column using a different part of the encryption key. This symmetric encryption is widely used in both hardware and software.

The output of the AES-256 algorithm is given below.

Key = 67EFD863A125D4469519F3CDC640156278EF90690110EC54
 Initialization Vector, IV = FAC2DAA667D3198F66DE01F932BCB58A

The SHA-512 hash algorithm is used in this work to ensure that the process data is not modified by the attackers. It takes the input data of length up to 2^{128} bits and produces 512-bit message digest as output. This message digest is based on all the bits in the input data and any change in the input bits would cause mismatch with its original message digest.

The output of SHA-512 hash algorithm in python is given below.

```
cf83e1357eefb8bdf1542850d66d8007d620e4050b5715dc83f4a921d36ce9ce47d0d13c5d85f2b0f
f8318d2877eec2f63b931bd47417a81a538327af927da3e
```

The output of the hybrid security algorithm includes asymmetric, symmetric and hash algorithm with the hardware key at the transmitter node is shown in figure 6.

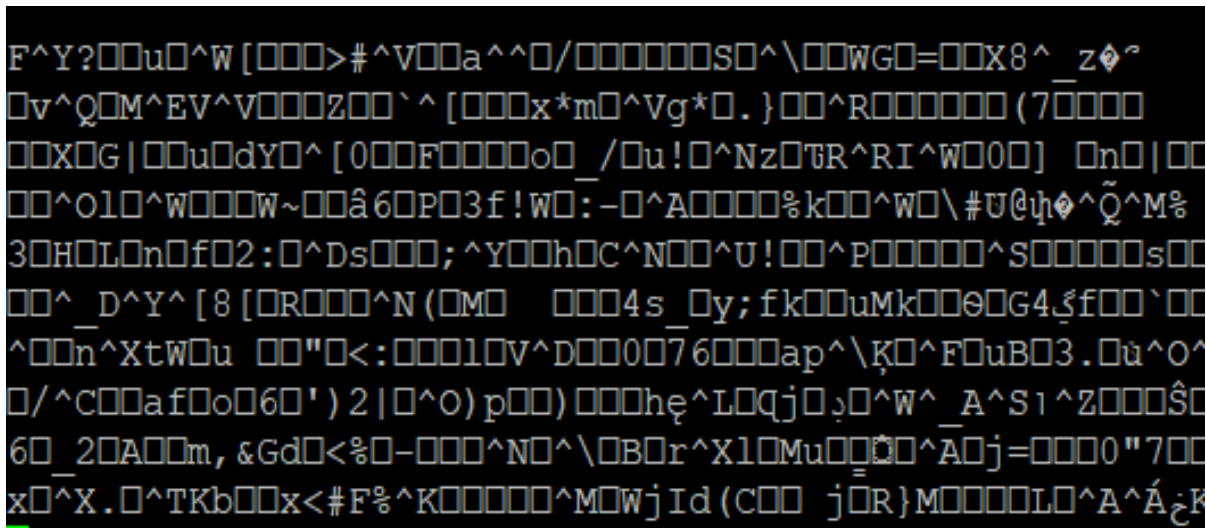


Figure 6. Output of the hybrid RSA, AES and SHA algorithm with Hardware key at the transmitter node

This encrypted data generated at the transmitter node is monitored and transmitted through the internet. At the receiver, two nodes are used to read the cipher text through the internet. The private key and the hardware key are essential to decrypt the cipher text and to obtain the process information in numerical value. The attackers are unable to access the encryption code and the encrypted data since it requires unique hardware key to view the code and the output. This proposed hybrid security algorithm with hardware key overcomes the drawback of existing security algorithms. It is suitable for protecting the sensitive plant information especially the hazardous processes and internet-based process monitoring and control applications. It enables the industrial operations to function very smoothly without any unauthorized modification of process parameters. This proposed security algorithm is the cost-effective method to

secure the process information and to provide safety to the process instruments as well as plant operators.

The comparison of proposed hybrid security algorithm with the existing security algorithm is shown in Table 1. The table indicates that the proposed hybrid security algorithm uses large key size as compared to the existing security algorithms. The security algorithm proposed includes hardware key and hybrid security algorithms developed in software whereas most of the existing security algorithms are performed using cryptography. The confidentiality, authentication and integrity of process information can be achieved with the combined features of proposed symmetric, asymmetric and hash function algorithms. The attackers are unable to access the hybrid security algorithms and the encrypted process data because the hardware key is required. The unique hardware key is included which allows only the authorized sender and receiver to access the sensitive process information.

Table 1. Comparison of proposed Security Algorithm with existing Security Algorithm

Authors	Security Algorithm	Key Size (Bits)	Number of rounds	Hardware Implementation	Security Level	Security Services
Sikhar Patranabis et al.[1]	AES	128	9	Yes	Medium	Data Confidentiality
Puneet Kumar et al. [3]	AES	128	16	No	Medium	Data Confidentiality
Mohamed Elhoseny et al., [4]	ECC and Homomorphic Encryption	176	-	No	High	Data and Image Encryption
RawyaRizk et al. [8]	AES, ECC, Dual RSA and MD5	128	10	Yes	High	Data Confidentiality, Integrity and Authentication
Proposed	RSA, AES and SHA with Hardware key	2048, 256 and 512	14	Yes (Monitoring through Internet)	Very High	Authentication, Data Confidentiality, and Integrity

The hybrid cryptography with large key size, hash algorithm, and usage of hardware key in this proposed work achieves very high level of security. This proposed hybrid security algorithm is performed and tested using embedded hardware in real time. It enables secure transmission and monitoring of sensitive process information through the internet.

D. Conclusions

Embedded systems are cost effective devices widely used in many process industries. The challenging task in designing the embedded system is to

incorporate security mechanism with embedded devices. This proposed hybrid security algorithm combines the features of symmetric, asymmetric and hash function algorithms with hardware key to ensure secure monitoring of process information through internet. The asymmetric algorithm with large key size of 2048-bit and the symmetric algorithm with key size of 256-bit are generated in this proposed algorithm which strengthens the security. The SHA-512 hash algorithm is used in this work which ensures data integrity during transmission of sensitive process data. The hardware key is essential to compile the hybrid encryption and decryption as well as to monitor the process data in cipher text and plain text. The hardware key is also encrypted along with the hybrid security algorithm. The hybrid encryption is performed at the master node and the hybrid decryption is performed at the two slave nodes. When the master node fails due to some fault, then the slave node1 acts as master and it sends the encrypted process data to the slave node2. The slave node1 acts as redundant node when master node fails as well as it performs hybrid decryption to generate the process data in numerical form when the master node works normally. It establishes secure transmission, monitoring and reception of plant information through the internet with the use of hardware key. It can be applicable for securing the plant parameters as well as process devices and it ensures smooth functioning of plant. It protects the sensitive process data from unauthorized access and provides safety to process instruments and plant operators.

E. References

- [1] SikharPatranabis, AbhishekChakraborty, Debdeep Mukhopadhyay, Chakrabarti, "Fault Space Transformation: A Generic Approach to Counter Differential Fault Analysis and Differential Fault Intensity Analysis on AES-like Block Ciphers," *IEEE Transactions on Information Forensics and Security*, vol. 12, Issue 5, pp. 1-11, 2017.
- [2] HaythamElmiligi, FayezGebali, M.Watheq El-Kharashi, "Multi-dimensional analysis of Embedded Systems Security", *Microprocessors and Microsystems*, Elsevier, vol. 41, pp. 29-36, 2016.
- [3] Puneet Kumar, Shashi B. Rana, "Development of modified AES algorithm for Data Security", *Optik Journal*, Elsevier, vol. 127, pp. 2341-2345, 2016.
- [4] Mohamed Elhoseny, HamdyElminir, AlaaRiad, Xiaohui Yuan, "A Secure data routing scheme for WSN using Elliptic Curve Cryptography and Homomorphic Encryption", *Computer and Information Sciences*, Elsevier, vol. 28, pp. 262-275, 2016.
- [5] Wei Jiang, Yue Ma, Nan Sang, ZiguoZhong, "Dynamic Security management for real-time Embeddedapplications in Industrial Networks", *Computers and Electrical Engineering*, Elsevier, vol. 41, pp. 86-101, 2015.
- [6] XiaojunZhai, Kofi Appiah, ShoaibEhsan, Gareth Howells, Huosheng Hu, DongbingGu, and Klaus D. McDonald-Maier, "A Method for Detecting Abnormal Program Behavior on Embedded Devices", *IEEE Transactions on Information Forensics and Security*, vol. 10, No. 8, pp. 1692-1704, 2015.
- [7] Giovanni Agosta, Alessandro Barengghi, Gerardo Pelosi, and Michele Scandale, "The MEET Approach: Securing Cryptographic Embedded Software Against

- Side Channel Attacks”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 34, No. 8, pp. 1320-1333, 2015.
- [8] RawyaRizk, Yasmin Alkady, “Two-phase Hybrid Cryptography algorithm for Wireless Sensor Networks”, Journal of Electrical Systems and Information Technology, Elsevier, vol. 2, pp. 296-313, 2015.
- [9] Apostolos P. Fournaris, Nicolas Sklavos, “Secure Embedded System Hardware Design – A flexible security and trust enhanced approach”, Computers and Electrical Engineering, Elsevier, vol. 40, pp. 121-133, 2014.
- [10] Wei Jiang, ZhenlinGuo, Yue Ma, Nan Sang, “Measurement-based research on cryptographic algorithms for embedded real-time systems”, Journal of Systems Architecture, Elsevier, vol. 59, pp. 1394-1404, 2013.
- [11] SatyajeetNimgaonkar, MahadevanGomathisankaran, Saraju P. Mohanty, “TSV: A Novel Energy Efficient Memory Integrity Verification scheme for Embedded systems”, Journal of Systems Architecture, Elsevier, vol. 59, pp. 400-411, 2013.
- [12] FatemehRaji, Ali Miri, Mohammad DavarpanahJazi, “Cryptographic Privacy Protection framework for Online Social Networks”, Computers and Electrical Engineering, Elsevier, vol. 39, pp. 2282-2298, 2013.
- [13] AbdalhosseinRezai, ParvizKeshavarzi, Zahra Moravej, “Secure SCADA communication by using a modified key management scheme”, ISA Transactions, Elsevier, vol. 52, pp. 517-524, 2013.
- [14] Hai Lin, YunsiFei, Xuan Guan, Zhijie Jerry Shi, “Architectural Enhancement and System Software Support for Program Code Integrity Monitoring in Application-Specific Instruction-Set Processors”, IEEE Transactions on Very Large Scale Integration Systems, vol. 18, pp. 1519-1532, 2010.
- [15] Austin Rogers, AleksandarMilenkovic, “Security extensions for integrity and confidentiality in Embedded processors”, Microprocessors and Microsystems, Elsevier, vol. 33, pp. 398-414, 2009.
- [16] DivyaArora, Srivaths Ravi, AnandRaghunathan, Niraj K. Jha, “Hardware-Assisted Run-Time Monitoring for Secure Program Execution on Embedded Processors” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 14, No. 12, pp. 1295-1308, 2006.