Microprocessors and Microsystems 72 (2020) 102921

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

Cryptographic system for data applications, in the context of internet of things

P. Panagiotou [a], N. Sklavos [b,c,*], E. Darra [b], I.D. Zaharakis [c,d]

[a] Delft University of Technology (TU Delft), Netherlands
[b] SCYTALE Group, Computer Engineering & Informatics Department, University of Patras, Hellas Greece
[c] Computer Technology Institute & Press "Diophantus" (CTI), Patra, Hellas Greece
[d] Electrical &Computer Engineering Department, University of Peloponnese, Hellas Greece

ARTICLE INFO

Article history:
Received 21 January 2019
Revised 13 September 2019
Accepted 18 October 2019
Available online 25 October 2019

Keywords:
Internet of things (IoT)
AES
UDOO Neo board
GCM
GMAC
One time passwords (OTP)
Two factor authentication
Security system

ABSTRACT

With each passing day, Internet of Things (IoT), has the potential to transform our society to a more digital way. In this paper, a cryptographic system is proposed, which has been designed and implemented, following the IoT optimized technologies. As the benefits of IoT are numerous, the need for a privacy platform is more than necessary to be developed. This work aims to demonstrate this by, firstly, implementing efficient and flexible, the fundamentals primitives of cryptography and privacy. Secondly, this is achieved, by introducing applied cryptography, in a more interactive and flexible approach. The proposed system and the incorporation of this platform is scrutinized. In the context of this work, an application of symmetric cryptography is introduced, based on the Advanced Encryption Standard (AES) in Electronic Code Book (ECB), Cipher Block Chaining (CBC) and Counter (CTR) modes of operation, for both encryption and decryption of texts, images and electronic data applications. In addition two other security schemes are supported by the proposed system: AES Galois/Counter Mode (GCM) and AES Galois Message Authentication Code (GMAC). The GCM proposed integration, in an authentication scheme, designed to provide authenticity and confidentiality, at the same time. On the other hand, GMAC, can be applied as message authentication code. Both operations, are optimized in sense of implementation resources, since the major cost is targeted to AES core. In addition, based on the integrated hardware modules, user registration and validation is proposed and implemented, with no additional cost, and with no performance penalty. Furthermore, two factor authentication has been designed and proposed, based on One Time Passwords (OTP), which can been produced with a random procedure. After these, a reference to the security levels, as regards to the communication between the IoT layers of the architecture, is presented. IoT hardware platforms are facing lack of security level and this brings the opportunity to use advanced security mechanisms. Implementation comparison results emphasize the importance of testing and measuring the performance of the alternative encryption algorithms, supported by hardware platforms.

1. Introduction

Nowadays, the development of Internet is being shaped mostly by instant data exchange. The pattern of device usage in an extensive and exhaustive way, is giving ability to the users, in order to communicate and share information. For this reason, the need for protecting the devices and the information dissemination among them, is determined as mandatory. Internet of Things (IoT), can be described as an application domain that integrates different technological and social fields [1]. Furthermore, IoT can bring together cryptographic models and security schemes, for implementation purposes, as described in [2]. For example, IoT systems can be based on AES, ECB and CBC modes of operation for symmetric encryption. Other algorithms such as RSA, SHA standards Diffie-Hellman and Elliptic Curve Cryptography (ECC), can serve as supplemented privacy schemes, for asymmetric cryptography.

Embedded systems design, are usually based on microprocessors, and microcontrollers. The role of embedded systems is to perform specific tasks such as sampling sensors and provide cost effectiveness and high portability, due to their small sizes. It is worthwhile mentioning, that these devices are discretized in how they interpret data. Moreover, they can be utilized for different applications as they are able to be programmable, flexible and can further be applicable to IoT environments. IoT can manage almost

* Corresponding author at: University of Patras, Computer Engineering & Informatics Department, University of Patras, 26504 Patra, Achaia, Greece.
E-mail address: nsklavos@ceid.upatras.gr (N. Sklavos).

https://doi.org/10.1016/j.micpro.2019.102921
0141-9331/

every aspect of appliances of everyday life. With an eye to the security lacks of various data applications, in IoT modern security systems are needed, in order to gain more strength, data processing capabilities, flexibility and technological evolution.

As the IoT technology offers a number of benefits to organizations, this invited paper, as an extension of our previous work [3] introduces a cryptographic system, specialized for data applications of IoT. It integrates alternative cryptographic and security services. The major target of the proposed system is to influence on the privacy systems, that are beneficial to both academia and industry needs, as IoT has gained increasingly more attention throughout the recent years. The cryptographic system can offer an interactive and trustworthy solution, in the way of selecting data texts, images and electronic data files, for privacy applications. In can been applied as a real time implementation, and to be used successfully, in both industry and academia sectors, or for individuals' needs. Furthermore, this work proposes a full IoT cryptographic system, that is based on UDOO NEO board [4], as distinct from various other works as proposed in [5–7], and focuses also on security and cryptographic applications of an IoT implementation system. This IoT system is based on Advanced Encryption Standard (AES) in various modes of operation [8]. More analytically it supports Electronic Code Book (ECB), Cipher Block Chaining (CBC) and Counter (CTR) modes of operation. It performs efficiently, for both encryption and decryption. It operates successfully for alternative users applications of text messages, image files of several types and electronic data applications, of all means.

Furthermore and compared to our previous work of [3], in this invited paper, more additional security mechanisms, are proposed and implemented. More analytically the introduced novelties of this work, which have been designed and implemented are:

1. Support of data encryption streaming, in addition to "original" block encryption of AES. This is achieved with no additional cost, based on Counter (CTR) Mode of operation. It is proposed for cases of not heavy duty data applications or user needs.
2. AES Galois/Counter Mode (GCM) and AES Galois Message Authentication Code (GMAC). GCM proposed implementation, integrates an authentication mechanism, which is aimed to support authenticity and confidentiality scheme, with no additional cost, in hardware resources, according to our research. On the other hand, the other proposed AES GMAC scheme, can be applied as message authentication code, trustworthy approach. GCM and GMAC, are proposed in sense of optimization, regarding available resources, since the critical component of the proposed system is AES core.
3. Registration and user validation, through hardware means. This can be achieved, through the push buttons of the current panel, with no additional cost, or performance sacrifice. Alternatively, dedicated push buttons can be used.
4. Two factor authentication of the user, after the successful login, of previous described process. This is proposed and integrated with one time passwords, (OTP), which are randomly generated.

Last but not least, the proposed cryptographic system could be enhanced and improved as to the users' requirements and expectations, in order to be expanded in a more accurate and efficient way.

This work has been developed under HORIZON 2020: UMI-Sci-Ed Project [9]. UMI-Sci-Ed Project (Exploiting Ubiquitous Computing, Mobile Computing and the Internet of Things to promote Science Education) aims at enhancing the attractiveness of science education and careers for young people via the use of latest technologies. We put Ubiquitous and Mobile Computing and the Internet of Things (UMI) into practice towards enhancing the level of STEM (Science, Technology, Engineering and Mathematic) education. At the same time, we are increasing the attractiveness of pursuing a career in domains pervaded by UMI for these youths [10].

UMI-Sci-Ed aims to empower youngsters to think creatively, apply new knowledge in an effective way, become continuously competitive in a highly demanding working environment. The ability to switch efficiently between different disciplines such as science disciplines depends on processing effectively the educational material based on clearly defined outcomes, expanding a broad repertoire of ICT communication, problem solving and decision-making skills, and using the collective knowledge represented in networks, based on working environments. The orientation of UMI-Sci-Ed is entrepreneurial and multidisciplinary in an effort to raise young boys' and girls' motivation in science education [10]. In UMI-Sci-Ed, technology itself is not starring as the objective of our work. Ubiquitous and mobile computing and IoT are rather used to support the UMI-Sci-Ed stakeholders working in education – educational community (teaching institutions, students, professors, tutors, etc.) and industry (UMI companies, VET providers, publishers, etc.) – career consultants, educational authorities and policy makers. To this end, communities of practice (CoPs) will be formed dynamically on the UMI-Sci-Ed platform around UMI projects implemented at schools, including representatives of all necessary stakeholders. In this project we aim to develop an integrated yet open training framework for upper high school students [9].

Technological institutions (CTI, CIT, CUBIT) and academic organizations (University of Helsinki, Norwegian University of Science and Technology and University of Pisa), are core participants and partners in UMI-Sci-Ed. ALL DIGITAL AISBL, (previously known as Telecentre Europe), a pan-European member association, supports the partnership in communication activities.

The structure of the paper is as follows: Section 2 gives the basic background of IoT technology. Section 3 gives the fundamental cryptography and privacy aspects. Section 4 describes the security level of systems and devices. The next section, presents in detail, the specified IoT board, of the proposed system. Section 6 is dedicated to the design, in detail, of the proposed cryptographic system, while the next section introduces, with full details, the implementation of the system. Implementations results and comparisons, follow. The last section, gives conclusions in a brief, and future directions of this work.

2. Internet of things technology: towards to a new era

As mentioned before, Internet of Things (IoT) refers to a network of physical devices (e.g. smart phones, laptops, smart watches etc.), that are connected to the Internet, collecting and sharing data [1]. Looking beyond the vast number of various applications, such as smart homes, smart cities, smart health infrastructure, one can realize and understand the level of digitalization, using such technologies. The level of device intelligence, could enable them to communicate without the involvement of a human beings, merging the digital and physical world. The rapid improvement of IoT can help people to make their environment smarter and more measurable, overcoming daily problems [11].

From the very beginning of IoT development, both academia and industry are targets of great importance. These applications can contribute in offering also innovative services and applications, implemented in different ways [12]. A huge number of applications, platforms and systems, can be deployed in order to help professors, researchers, industry experts and manufacturers to work hard, for a qualitative content in each and every device. This content can be customizable and extended to the level of maturity that different technologies are connecting.

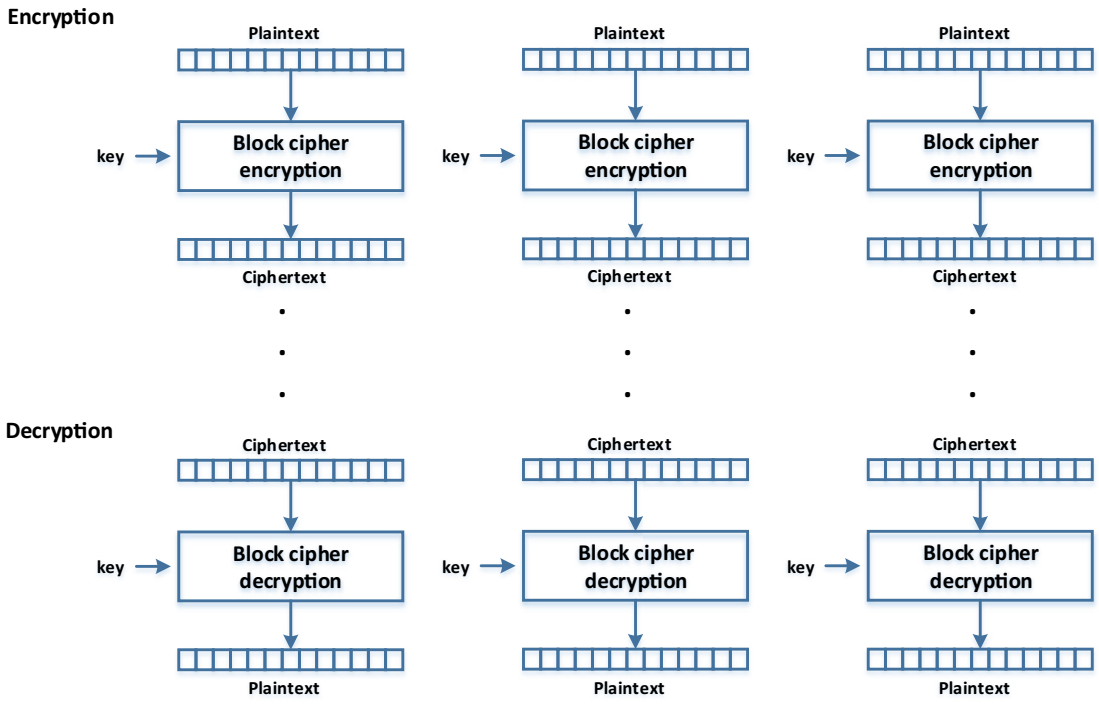

Fig. 1. Electronic code book (ECB) mode of AES.

As technology changes humans' lives in a continuous manner, the physical devices have the advantage to be connected and alternatively used. For that reason, both industry and academia can use methodologies to minimize, if not eliminate their operational costs. Additionally, approaches of using or even reusing portable devices, is completely acceptable. The devices can be connected to the current technology, in order to develop particular systems with low cost and high performance. This novel practice gives the advantage to the people, to connect from all over the world, through a hardware system, that is compatible, stable and safe.

The proposed cryptographic system, can been applied successfully as a real time platform for everyday use. Additionally it can be used for research purposes, and training methodologies or educational activities, in both industry and academia, sectors.

3. Privacy and cryptography primitives

In this section, we give the fundamental primitives and definitions of cryptography, as well as of symmetric key block ciphers, for both encryption and decryption processes. Symmetric key cryptography is called the operation, when the same key is used for both encryption and decryption. With this process where the sender encrypts the plaintext using a key and a selected block cipher. In this way, the ciphertext is produced. On the other side, the receiver decrypts the ciphertext using both same key and the block cipher. The key is exchanged securely, as it is transferred in a secure channel before the data exchange. In this phase, a malicious insider (adversary) is trying to monitor the communication and/or alter the data, but is unfeasible due to the known attack methods, assumed that a large, in wide, key is applied.

A block-by-block, way of operation is used by symmetric block cipher in order to process data. The size of each block is fixed to n-bit, such can be $n = 128$-bit. If the plaintext contains more than one n-bit blocks, there are modes of operation, that additionally supports, the original operation of a block cipher. In the context of this work, three modes of operation are presented; Electronic Code Book (ECB), Cipher Block Chaining (CBC) and Counter (CTR) modes [13]. Advanced Encryption Standard (AES) transforms data blocks of $n = 128$-bit and is implemented for the symmetric block cipher mode.

The most widely used mode is ECB (Fig. 1), and breaks the plaintext into i-blocks and encrypts each block separately. Each block that refers to the plaintext is encrypted, using a key and produces a ciphertext-block, independently. On the other hand, in the decryption process the inverted encryption process takes place. Each ciphertext-block is decrypted using a key and produces the plaintext.

In the CBC operation mode, as presented in Fig. 2, an Initialization Vector IV is used in the first block, of encryption process. The IV uses an exclusive-OR logic function, in order to be XORed with the first plaintext block. The result is encrypted and in this way, the first ciphertext block is produced. Then, each plaintext block is XORed with the ciphertext output of the previous stage. In the opposite process, for the decryption of a ciphertext block, the ciphertext block uses the XOR function with the previous stage block, and the plaintext is produced. We should note, that in this case, the IV is used only once at the beginning of the decryption process, using the XOR operation, with the first generated decrypted block of the first ciphertext block, since there is no previous stage. In this way the first plaintext block is produced.

In ECB and CBC modes of operation, if the final produced plaintext is equal to the initial one, for every input block, then both processes (encryption and decryption) are successfully completed. This assumes that the same key is used for both of them. However, as it is illustrated in ECB mode, the encryption of identical plaintext blocks, produces identical ciphertext blocks and vice versa. This is the most serious disadvantage of this mode, as all blocks are encrypted independently and all used messages should not be greater than one block. In CBC mode, this mentioned disadvantage does not exist. It makes use of the mechanism of encrypting each plaintext block with the previous block (chaining mechanism). An advantage of CBC mode is that different use of IVs, but with the same key in the first phase of encryption, results finally to different ciphertexts.

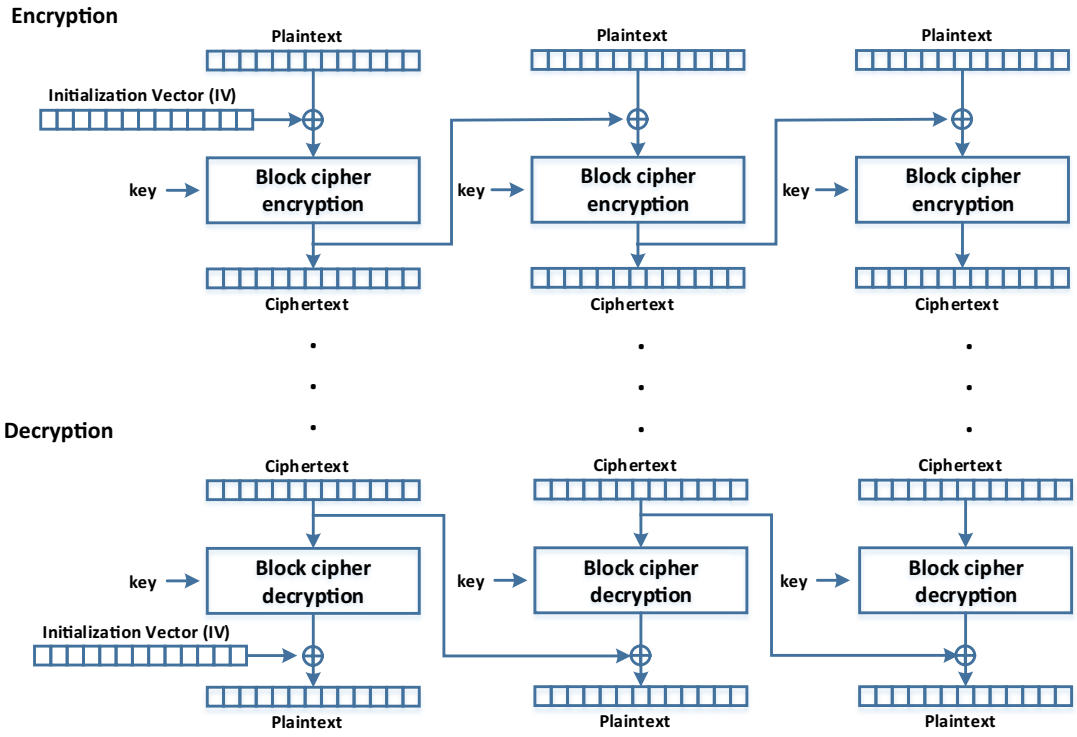

Fig. 2. Cipher block chaining (CBC) mode of AES.

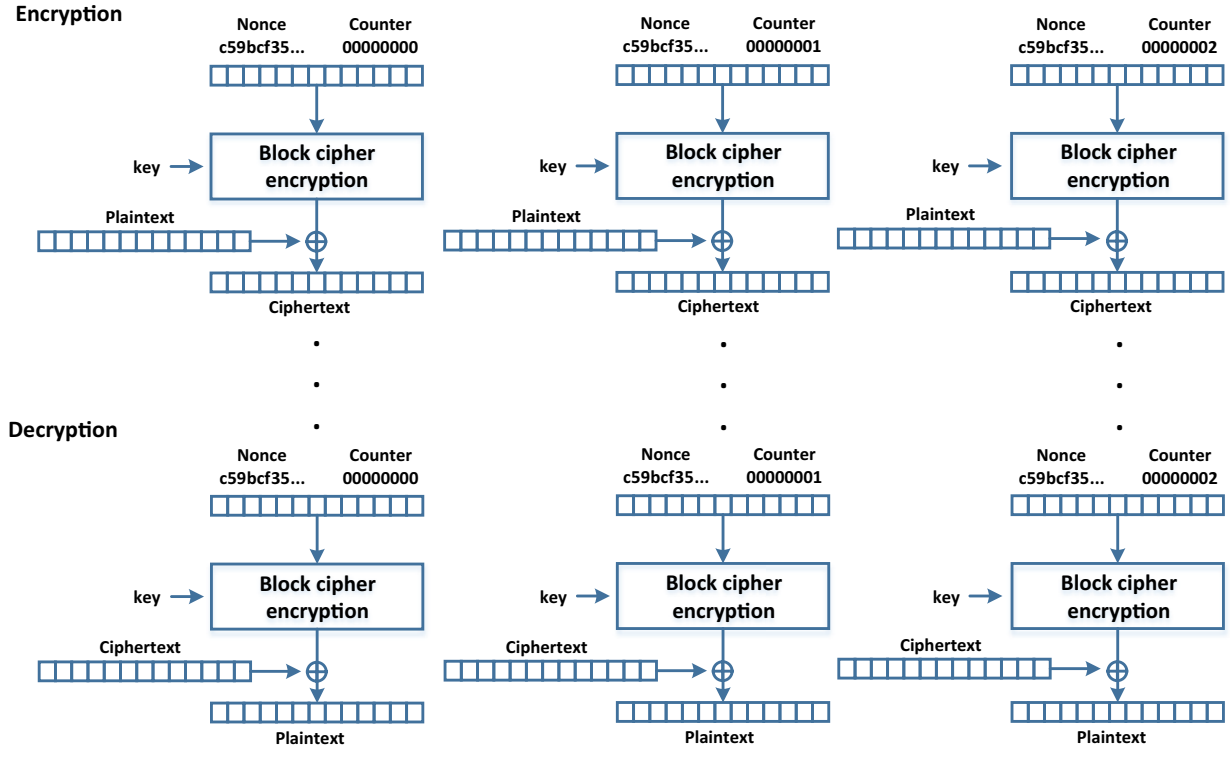

Fig. 3. Counter (CTR) mode of AES.

In addition to ECB and CBC modes, with CTR mode of operation (Fig. 3), a block cipher operates as a stream cipher, at the same time, with no additional cost. It is able to generate the next keystream block, by encrypting values of a "counter". The counter is a function that provides a sequence number, that is not repeated for long. In CTR mode the nonce value is random and is also similar to Initialization Vector (IV). It can be combined with the counter, using for example an XOR function, in order to produce a unique counter block for encryption. In case that there is a non-random nonce, both the nonce and the counter should be connected by simply ADDing or XORing them into a single value. Once an attacker controls the counter pair and the plaintext,

XORing the ciphertext with the known plaintext would yield a value that, when XORing with the ciphertext of the next block, would decrypt the block.

4. Security level and implementation devices

One of the main concerns regarding IoT networks is the security level as regards, to the communication between the IoT layers of the architecture. The security level is used mainly in symmetric and asymmetric cryptography, in order to measure in bits, the strength that a cryptographic primitive achieves. It is worth pointing out that IoT hardware platforms are facing lack of security level. This arises the crucial need to use advanced security mechanisms and privacy schemes. In addition, one of the most critical issues in IoT applications is the power efficiency, as the devices have to operate using energy sources of battery. Later on, we describe the algorithms found in the corresponding literature, that make use of different implementation devices, that adopt different infrastructure.

Authors in [5], use advanced security mechanisms, such as Transport Layer Security (TLS). RSA and Elliptic Curve Diffie–Hellman Ephemeral (ECDHE) are defined as the most popular and recommended cipher suites for TLS. The cipher suites, based on RSA algorithm, make use of it for key-exchange. In contrast, ECDHE-based cipher suites, use Ephemeral Diffie-Hellman based on Elliptic Curves cryptographic algorithm. The ESP32 embeds an IEEE 802.11b/g/n interface and supports Bluetooth v4.2. The core of the SoC is a 32-bit LX6 dual-core microprocessor that operates at up to 240 MHz, with 520 KB of SRAM. The hardware acceleration engine for cryptographic algorithms supports AES, SHA-2, RSA, ECC and also presents a Random Number Generator (RNG). It can be powered by either a 3.3 V or a 5.0 V power source, or by using a micro-USB connector.

In [6], a Field Programmable Gate Array (FPGA)-based implementation of Advanced Encryption Standard (AES) and Secure Hash Algorithm-3 (SHA-3) algorithms, is described. This proposed architecture allows both data integrity and confidentiality to be provided for high-speed IoT applications. The security level of hash functions is proven more complicated, to the way that it is needed, in order to provide high speed and real time results. The implementation results on Artix-7 FPGA are better in the sense of low power operations as compared to other FPGAs. The functionality of SHA-3 is implemented using LUT-6 primitives and they are instantiated for the complete implementation of SHA-3. FPGAs) are made up of an interconnection of logic blocks in the form of a two-dimensional array. The logic blocks consist of look-up tables (LUTs) which are constructed over simple memories that store Boolean functions. Each LUT consists of a fixed number of inputs and is coupled to a multiplexor and a Flip-Flop in order to build sequential circuits.

As described in work [7], AES is an identifiable cryptographic algorithm that can be used to protect electronic data. AES is a symmetric block cipher algorithm with block length of 128-bit, and generally allows three different key lengths 128-, 192- and 256-bit, respectively. Authors of work [7] propose an implementation of AES on ARM Cortex-M3 processor with minimum memory that will be useful for deploying it in low cost applications such as IoT. LPC1769 development boards are supported by 512 KB of flash memory, and 64 KB of data memory. It can run a Cortex-M3 core at up to 120 MHz. LPC1769 belong to Cortex-M3 family of 32-bit processors by ARM. These microprocessors have 16 32-bit registers, of which three are reserved for program counter, stack pointer, and link register.

Although IoT devices have many advantages in terms of scalability and cost, they are restricted in terms of computing capabilities and hardware resources. So it seems to be impossible, to implement complex and heavy operations needed by encryption algorithms, to cipher and secure the communications. However, it should be cited that the above mentioned implementation devices, could be useful for the scope that are able to accomplish, but in some cases there limitations in sense of performance, storage, autonomy and other similar capabilities.

5. Selected IoT board

The UDOO Neo board [4] is used as a basis to implement a full IoT solution, in order to combine different individual modules, for the proposed system. It is a low cost, series board, with good performance, enough storage capabilities, flexible autonomy, and good energy consumption. These characteristics are proven efficient for our research, always keeping a good balance between them. This balance, is a major criterion of UDOO Neo board selection, since our research does not need or give a special priority to any of them. In this way, fair and satisfactory results can been achieved, compared with previous other works and approaches, which have too specified designed criteria, or other priorities and dedicated needs [5–7]. Last but not least, the proposed system design and implementation, are not depended in any special characteristics, of the selected board, and can be applied successful also to similar ones, according to user's need, availability and decision, each time. This is proven as another major advantage of the proposed cryptographic system.

More analytically, a single board computer was lately launched, based on Arduino with Android or Linux, enriched with sensors, Bluetooth 4.0 and a Wi-Fi module [4]. At the beginning, it was used as a training testbed low cost system, where several new applications and services can be developed [11,12].

In detail, the specifications of UDOO Neo board are the following: an NXP/Freescale iMX 6SoloX processor, that has two heterogeneous processors embedded on the same chip, an ARM Cortex-A9 and an ARM Cortex-M4 embedded processor, with clock at 1 GHz and 200 MHz, respectively [14]. These two processors communicate through a virtualized serial interface, that uses the shared memory to exchange data. The processors, also, share hardware implemented features, provided by the architecture. The iMX 6SoloX is connected to the peripherals of sensors, Bluetooth 4.0 and Wi-Fi module. The peripherals are connected to processors, through a high speed AXI bus, using different hardware interface (I2C's, SPI, GPIOs, UARTs and others).

All hardware features of UDOO Neo board, can be accessed and connected via processor's pad, with an editable mixing. Therefore, the functions are not fixed, but can be accessed on different pads. Some of these are connected to the external pins, for both processors, allowing the users to connect their own peripherals. General Purpose Input/Output (GPIO) pins, can be dynamically shared, at boot time, between the Cortex-A9 and Cortex-M4 processors.

6. Proposed cryptographic system

A novel cryptographic system is proposed, designed for IoT technology, and is presented in detail. It makes use of the UDOO Neo board [4]. The implementation board is based on and uses a different, than usually defined operating systems, named UDOObuntu 2, that is based on Ubuntu 14.04, without any Graphical User Interface (GUI). The proposed software architecture is illustrated in the Fig. 4. The Wi-Fi module operates as an access point, based on protocol IEEE 802.11 b/g/n. The device can handle up to 10 clients, to this mode.

The UDOO Neo board is constituted as a web application that consists of Apache and PHP modules. It can also handle Hypertext Transfer Protocol (HTTP) requests, from web browsers. The interface has been developed with web technologies, such as Hyper-

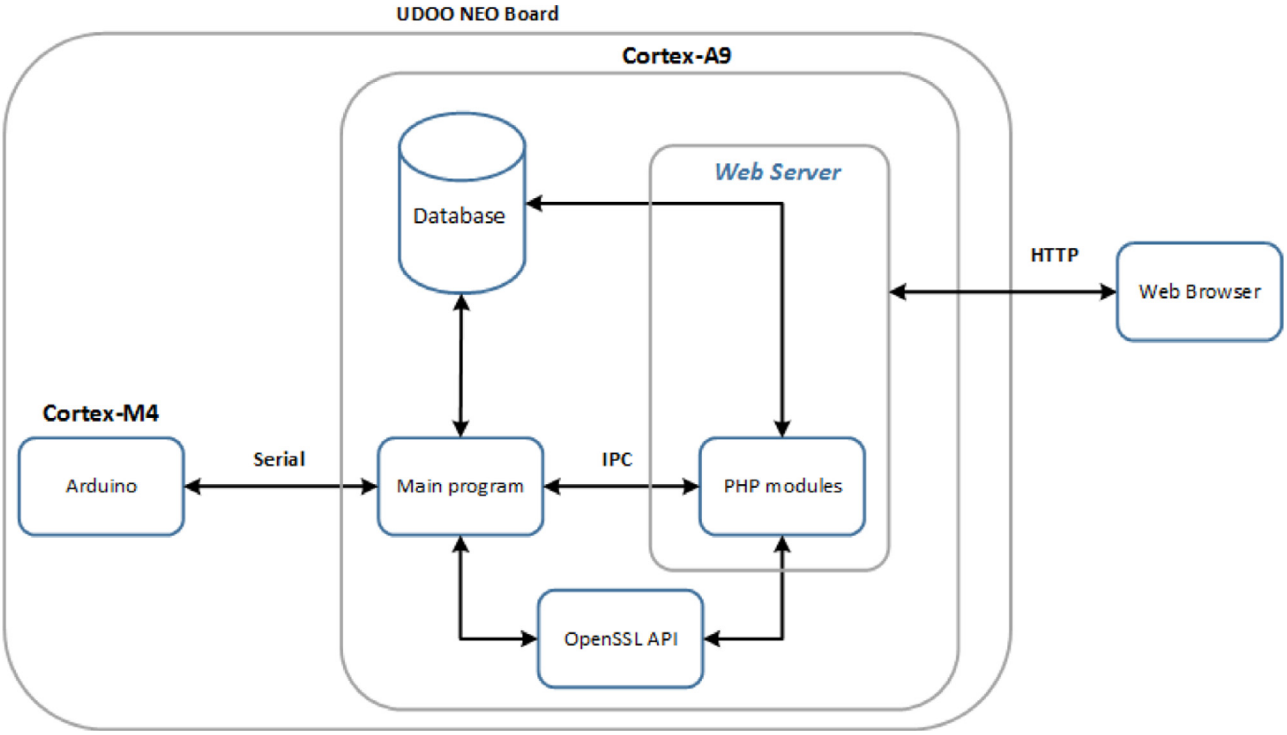

Fig. 4. Proposed system architecture.

Text Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript programming language. The user gains access to these technologies, through the web browser, as well as, communicates through the Wi-Fi protocol.

It is worth mentioning that the system has been implemented using the OpenSSL Application Programming Interface (API) with C/C++ programming language [15]. The API contain information about the interface, encryption or decryption requests like the HTTP requests. The process is the following: at first a request (either encryption or decryption) is sent to OpenSSL module, then synchronous or asynchronous cryptographic results are expecting to arrive in the time being. Python is used to interpret the function of the tool. It also performs precise commands to save date and other operations. Python program communicates through the interface of Cortex-M4, and Inter Process Communication (IPC), to PHP module. The use of a database is proven absolutely necessary for data storage.

Going further to the Cortex-M4, we can indicate that an Arduino based code, is running. This code is responsible for handling all the external peripherals such as a button that should be pushed to indicate the appropriate status of RGB leds, in case of reporting sensor data on a screen. All interaction and reporting, are then passed through bidirectional serial communication, to the main function of the tool.

In Fig. 5, we can clarify the development of the proposed system, using UDOO Neo board and peripherals. One of these peripherals is a TFT Color Display ILI9163, which can be used optionally and only in case of monitoring purposes [16]. It is connected through a Serial Peripheral Interface (SPI) bus and a DHT11 humidity and temperature sensor, connected through a serial interface using a single-wire two-way protocol. In addition, an RGB led is connected to output pins with Pulse Width Modulation (PWM) mode [17]. The two push-buttons are connected to input pins, with pull up resistors and handled by interrupts.

A person can be a registered user of the proposed system, through a registration and validation security scheme, based on integrated hardware modules of the proposed system (Fig. 6). More analytically this is implemented, with two push buttons of the panel. Through the registration process a user is informed for the remaining time of the process (Fig. 6a), the successful validation result (Fig. 6b), possible timeouts of performed operation (Fig. 6c) or even the unsuccessful attempt (Fig. 6d), especially in case of hacking or possible attacking.

Two factor authentication of the user, is also implemented and supported, as a crucial authentication scheme. It follows after the successful login, of the above described process. The applied technique that has been adopted is the one time passwords, (OTP).

The last ones are randomly generated, by means of the device and no additional cost. Accelerometer is used in our case, in order to optimize the seed of a common pseudo-random bytes, function, implemented with a software routine. In this way, better “random number” are produced, in sense of vector range, guess probability etc. First, the appropriate number of bytes is selected, as well as the level of sensor sampling: time slots, range of produced vector etc. Then the accelerometer is moved in a random way by the user, and the “random” values are sampled. The process is supported with a the ability of cancelation, in case of not desired result. When the progress of the process is completed, 100%, the random bytes have been generated to the output (Fig. 7).

7. Proposed IoT system implementation

In this section, the proposed IoT system implementation is presented. The first step contains the connection of the user device to the proposed IoT system, through the Wi-Fi access point. In the next step, a web browser is used to gain access to the supported services.

Then, the user creates an account and registers to this with his/her preferred e-mail address and password. A validation of the account follows in a physical way using the two push-buttons of the device. After this, another validation process is performed: a two-factor authentication. In this phase, the IoT device generates

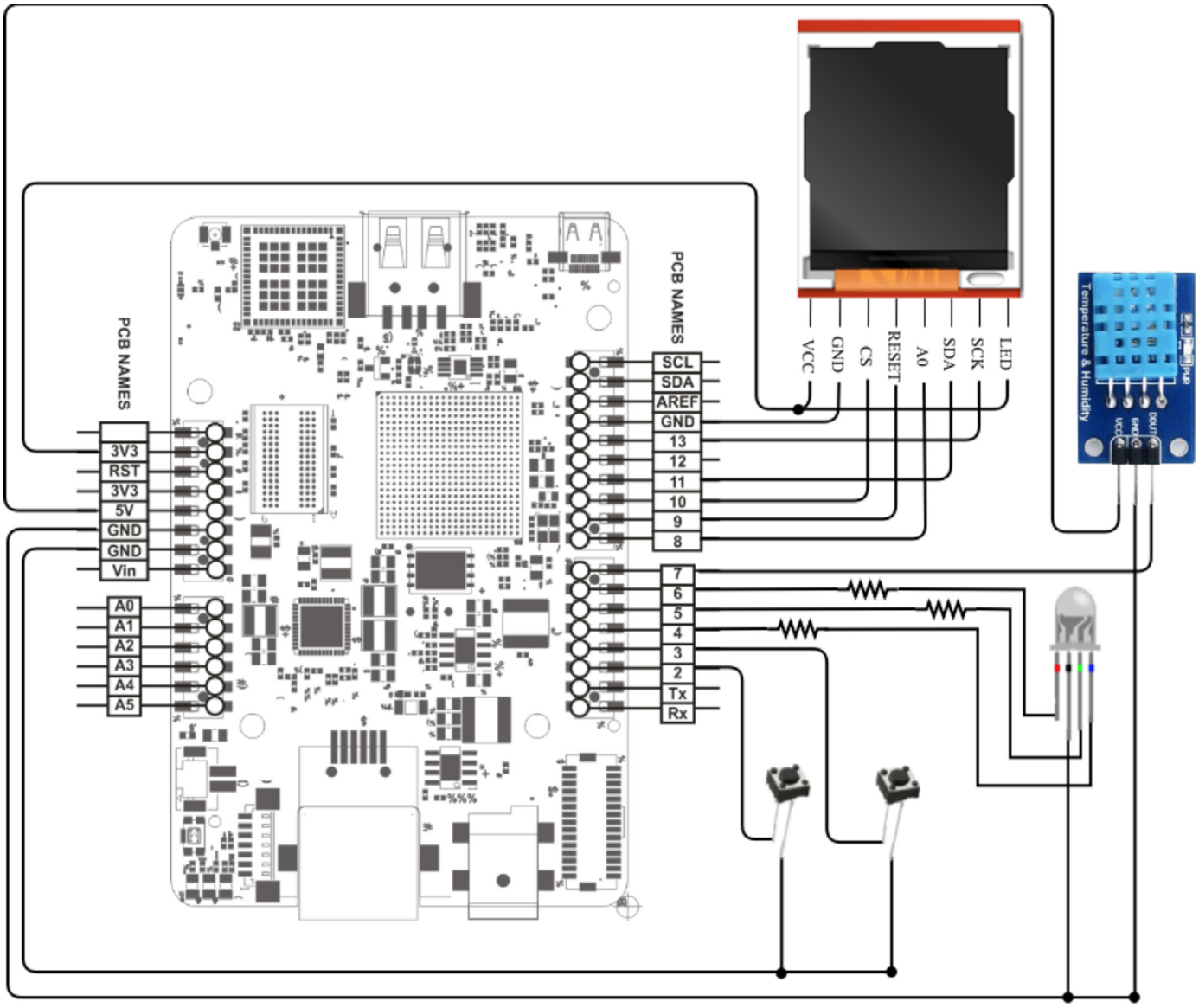

Fig. 5. Proposed system: implementation device and peripherals.

Fig. 6. User's validation module and two factors authentication scheme.

randomly an One Time Password (OTP). This password is equal to four characters and it can be displayed on the screen. The user enters this OTP, in order to have access to the decrypted data of the system.

When the user gains access to the interface of the system, he/she can perform encryption or decryption, as defined in the previous Section 3, using AES [8], on three different types of applications:

1. Data Text: User input text.
2. Image: Image known file formats (bmp, png, jpeg etc.).
3. Electronic Files: All the other data file formats.

Fig. 7. Random numbers generation.

Further to this, various privacy schemes can be selected by the user to perform encryption and/or decryption. The supported privacy schemes by the proposed system are the followings:

1. "Original" AES
2. AES Galois/Counter Mode (GCM)
3. AES Galois Message Authentication Code (GMAC)
4. Modes of Operation (ECB, CBC, CTR)
5. Key length (128-, 192-, 256-bit)
6. Padding (No padding, PKCS#7 padding), [18]
7. Password (Key is generated using a Key Derivation Function – KDF)
8. Key – using hexadecimal notation. The size depends on key length
9. IV (only in CBC mode) – using hexadecimal notation and is equal to block size (16-byte).

As it has been mentioned above, the proposed system, in addition to the confidentiality that a) "original" integrated AES achieves, proposes and implements b) AES Galois/Counter Mode (GCM) and c) AES Galois Message Authentication Code (GMAC), as are shown in detail in the following Fig. 8.

According to this research, AES GCM implementation, has been integrated as an authentication security scheme, which is targeted to provide authenticity and confidentiality process, with no additional cost, in hardware terms. On the other hand, the introduced AES GMAC integration, is proposed for message authentication code, as a flexible and trustworthy solution at the same time. Both AES GCM and GMAC, are proposed in sense of the available resources optimization, due to the fact, that the additional hardware cost is minimal, compared with the integrated AES, main module.

As it has been mentioned before in this work, an example will follow to indicate the critical disadvantages of ECB mode of operation. This example is based on the proposed system supported services. The scenario applies an image which is converted into a BMP file format. The header of this encryption is extracted. The body of the image is encrypted using the AES and the user's privacy schemes. Then, the encrypted body is linked to the header. The whole encrypted image is shown in the next Fig. 9. A user can encrypt the example image. The selected example image is the official logo, University of Patras, (our institute). The converted image has a fixed header of 138 bytes and a body of 1,866,400 bytes. AES encrypts 1,866,400/16 bytes (block size) = 1,166,500 blocks and there is no need for padding for the selected image. The parameters used for the image encryption are: AES encryption, ECB mode of operation, key length of 128-bit, no need for padding and an initialization vector equal to "00000000000000000000000075BCD15". The image as shown below uses the same pattern of data. This means that the data blocks that are encrypted, follow the same pattern and exploit the image. The use of key, the specified length and the password phrase are not mutually reliant, and the image will be exploited in any way.

As another encryption example, one must be very careful when multiple encryptions of the image are performed. The result of the encrypted image will not be "correct", as illustrated below in Fig. 10. This image is encrypted three times, using the same previous AES parameters: $AES_{ECB}(AES_{ECB}(AES_{ECB}(image)))$.

The applied image is also used to be encrypted, selecting the same parameters but using CBC operation mode. The key used is the same as before and the IV is selected to be equal to zero. As depicted in Fig. 10d, the encrypted image with CBC mode encrypts the image successfully by removing all previous patterns. In the CBC mode of operation, the advantage is that a possible adversary cannot identify the image that has been encrypted.

While using another high quality image as an example of encryption, based on the proposed system, it is perceived that it was used the AES in ECB mode with the same key. The result of the encrypted image is illustrated on the right part, of Fig. 11. The most important issue is that the encrypted image, cannot be identified. If we want to be more accurate and examine the encrypted image closely, we could find same patterns in this, as well. If we can be more precise to the encrypted image and focus on its bytes, an adversary can discern same ciphertext blocks, that will lead to

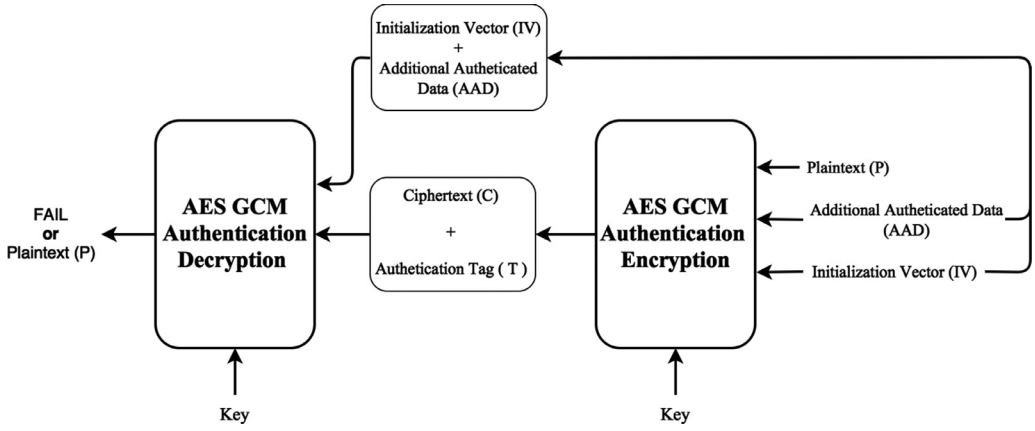

Fig. 8. AES authentication schemes.

Fig. 9. (a) original image, (b) encrypted image using AES-ECB mode.

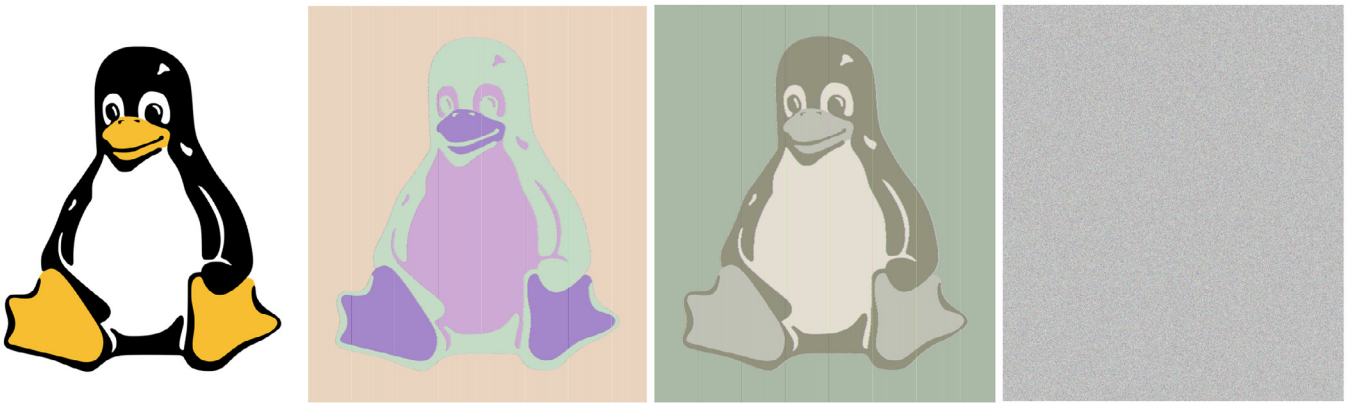

Fig. 10. (a) Linux "Tux" original image, (b) one encryption AES-ECB, (c) multiple encryptions, AES-ECB, (d) Encrypted using AES-CBC.

Fig. 11. (a) original high quality image, (b) encrypted image using AES-ECB mode.

identical blocks of the original image. As a result, the CBC encryption mode can be defined as more accurate and flexible, to give solution to the encryption issue presented previously.

Both encryption and decryption of ECB mode, can be easily parallelized using multiple AES cores. On the other hand, the encryption of CBC mode cannot be performed in parallel, because the encryption of a plaintext block, depends on the ciphering, of previous plaintext blocks. This chaining dependency cannot be done in parallel, thus only the AES cipher can be parallelized. In contrast, the decryption process of CBC mode can be parallelized due to the non-dependency of previous decrypted blocks.

In the following Fig. 12, we compare the number of clock cycles per block between non-parallel encryption with ECB and CBC mode of operation that the UDOO Neo board needs. In x-axis, the number of blocks that will be encrypted is shown while in y-axis the clock cycles are presented. As it is depicted from the graph, the ECB mode of operation, spends fewer number of clock cycles than the CBC mode needs. This is due to the fact that the CBC mode performs, for every encryption of a plaintext block, the XOR function.

If we want to make parallelism between these two encryption modes, the results will be proven different. Fig. 13 presents that for a small number of blocks, the ECB mode can be parallelized successfully, and the needed clock cycles of these blocks encryption, are continuous. However, the CBC mode cannot be parallel and spends much more clock cycles, than before.

8. Implementations results and comparisons

In this section, we compare the results extracted from different ciphers, previously introduced. We can distinguish these implementations in two types: as regards to symmetric and asymmetric cryptography.

As for the symmetric cryptography we can mention the implementation proposed in [7], that uses the LPC1769 development board with 512 KB of flash memory and 64 KB of data memory and our proposed one that uses the UDOO Neo board that implements AES algorithm with a 128-bit key and the ECB and CBC modes of operation, (Fig. 14).

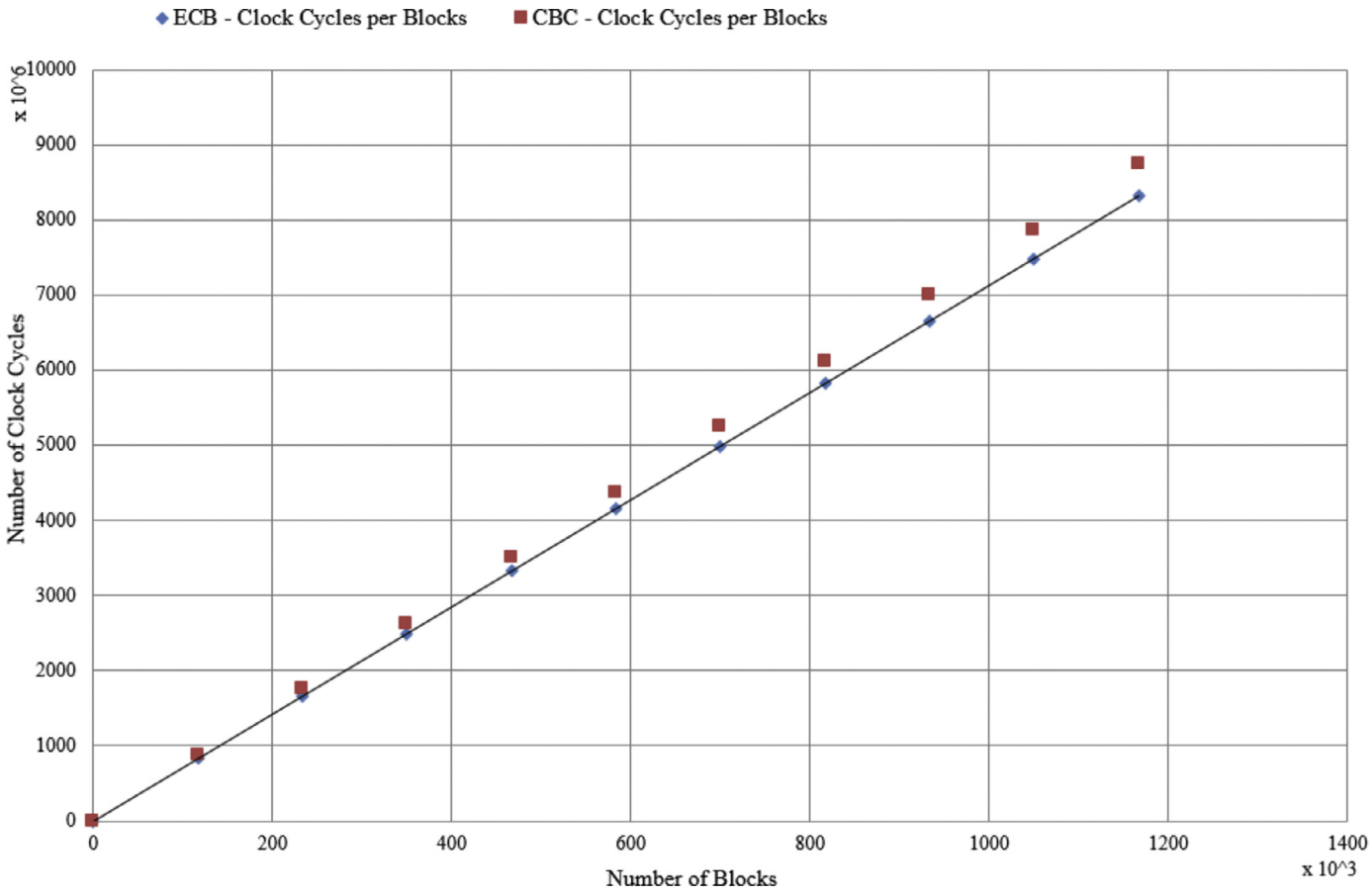

Fig. 12. AES performance: clock cycles/Blocks: Not parallel implementation.

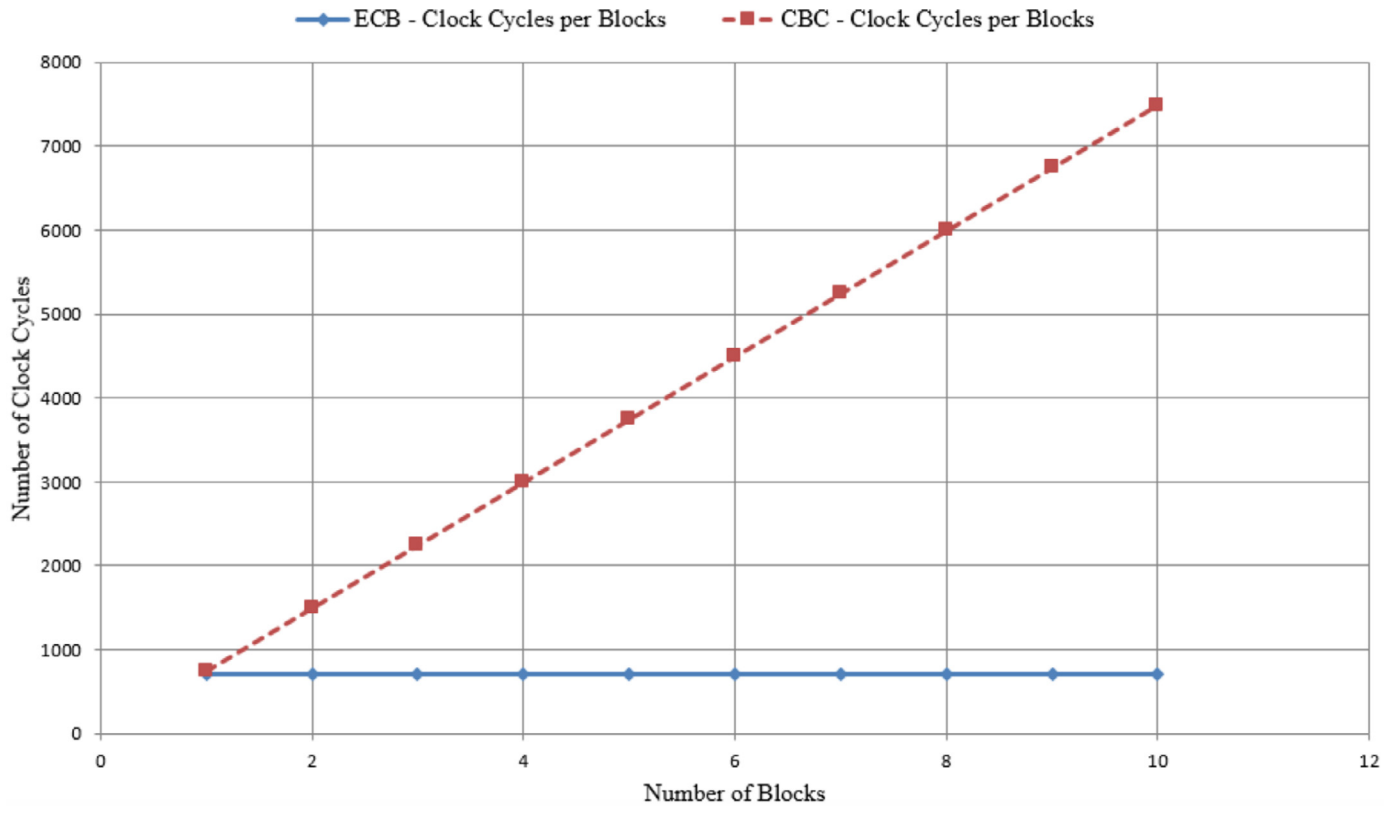

Fig. 13. AES performance: clock cycles/Blocks: Parallel implementation.

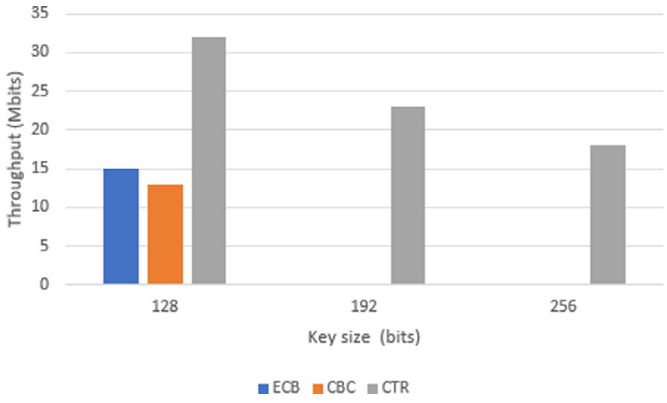

Fig. 14. Symmetric cryptography comparison graphs.

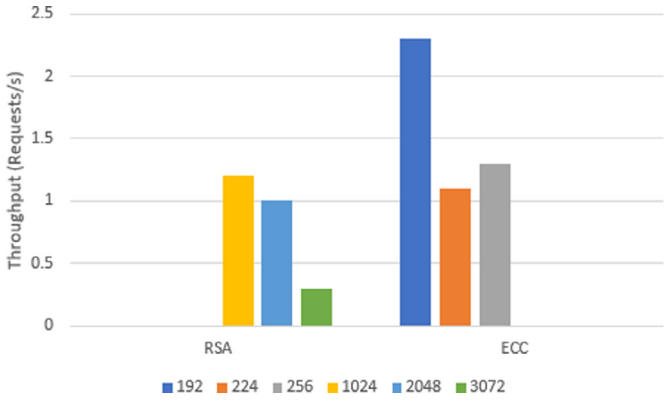

Fig. 15. Asymmetric cryptography comparison graphs.

In asymmetric cryptography in order to compare ECC and RSA cipher suites, it must be taken into account their security level, (Fig. 15). Such a level is a value that quantifies the required effort to break a cryptographic primitive. If the effort is 2k, it is said to offer k-bit security and, therefore, it provides a security level k. For example, a 128-bit security level is achieved by either using 3072-bit RSA or just a 256-bit key size of an ECC curve. The obtained results show that the curve secp256r1 outperforms the curve secp224r1, while providing a higher security level, due to optimizations made on the libraries implementing the ECC operations.

The obtained results emphasize the importance of testing and measuring the performance of the different algorithms supported by hardware platforms. The created testbed allowed for accurately comparing the different alternatives in terms of security.

9. Conclusions & outlook

The need of a trustworthy cryptographic system, that exploits and implements the IoT technology, is more than necessary, in order to have a flexible connection between the physical and virtual worlds. For these reasons, a novel IoT cryptographic system is proposed in this work, offering a vast number of security schemes. An approach of the different security levels, in terms of encryption/decryption and their efficient implementation, in IoT device are introduced. Furthermore, a brief reference to the security levels as regards the communication between the IoT layers of the architecture, are studied in details.

The proposed cryptographic system, as an extended invited work of the our preliminary publication [3], can be extended more, as a more sophisticated design and powerful system; the objective could be to expand the system by implementing other cryptographic primitives, like Public Key Cryptography, Hashing, and Digital Signatures. The area lightweight cryptography, such as stream ciphers, could also be considered [19,20], as well as cryptography and security systems, for the sensitive area of health and medical applications, can be also crucial future directions [21–23]. Additionally, this system should provide a connection to multiple boards using the same platform technology, for information exchange and suggest solutions to different security problems [19–21].

Declaration of Competing Interest

According to our knowledge no conflict of interest with the possible reviewers is applicable in our case, besides people affiliated with the same, authors' institutes (or country).

Acknowledgment

This work is under the UMI-Sci-Ed (Exploiting Ubiquitous Computing, Mobile Computing and the Internet of Things to promote Science Education) project. This project has received funding from the European Union's HORIZON 2020 research and innovation program under grant agreement No 710583.

References

[1] Towards a Definition of the Internet of Things (IoT) May 2015 IEEE Internet Initiative Revision 1, Published 27.
[2] Fei Hu, Security and Privacy in Internet of Things: Models, Algorithms, and Implemenations, CRC Press, Taylor Francis Group, ISBN: 978-1-4987-2318-3, 2016.
[3] P. Panagiotou, N. Sklavos, I.D. Zaharakis, Design and implementation of a privacy framework, for the internet of things (IoT), in: Proceedings of 21th EUROMICRO Conference on Digital System Design, Architectures, Methods, Tools (DSD'18), Prague, Czech Republic, 2018 August 29–31.
[4] UDOO Neo Board Specifications March 2018 Available at: https://www.udoo.org/.
[5] M. Suarez-Albela, T.M. Fernandez-Carames, P. Fraga-Lamas, L. Castedo, A practical performance comparison of ECC and RSA for resource-constrained IoT devices, Global Internet of Things Summit (GIoTS)„ May 04, 2018.
[6] M. Rao, T. Newe, I. Grout, Secure hash algorithm-3 (SHA-3) implementation on Xilinx FPGAs, suitable for IoT applications, in: Proceedings of the 8th International Conference On Sensing Technology, Liverpool, UK„ Sep. 2-4, 2014.
[7] R.W. Wardhani, D. Ogi, M. Syahral, D. Septono, Fast implementation of AES on Cortex-m3 for security information devices, 15th International Conference on Quality in Research (QiR): International Symposium On Electrical and Computer Engineering, July,2017.
[8] William Stallings, Cryptography and Network Security: Principles and Practice, sixth ed., Prentice Hall, 2013.
[9] HORIZON 2020, UMI-Sci-Ed Project (Exploiting Ubiquitous Computing, Mobile Computing and the Internet of Things to Promote Science Education), 2016–2019, March 2018 , Homepage http://umi-sci-ed.eu/ .
[10] I.D. Zaharakis, N. Sklavos, A. Kameas, Exploiting ubiquitous computing, mobile computing and the internet of things to promote science education, in: Proceedings of 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS'16), Larnaca, Cyprus, November 21-23, 2016.
[11] S. Theodorou, N. Sklavos, Blockchain based security & privacy in smart cities, in: D.anda B. Rawat, K.ayhan Z. Ghafoor (Eds.), Chapter in the Book: Smart Cities Cybersecurity and Privacy, Elsevier Press, 2019 ISBN: 9780128150320.
[12] S. Zeadally, A.K. Das, N. Sklavos, Cryptographic technologies and protocol standards for internet of things, Internet of Things: Engineering Cyber Physical Human Systems, Elsevier Science Press, 2019.
[13] NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation—Methods and Techniques (December 2001).
[14] NXP/Freescale iMX 6SoloX, "i.MX 6Solo/6DualLite applications processor reference manual", March 2018.
[15] OpenSSL - Cryptography and SSL/TLS Toolkit Available at: March 2018, https://www.openssl.org/.
[16] ILI9163 1.44 Inch TFT LCD Display, a-Si TFT LCD Single Chip Driver 132RGBx162 Resolution and 262K Color, Documentation, 2018, available at: https://www.rockbox.org/wiki/pub/Main/SonyNWZE370/ILI9163.pdf, March 2018.
[17] DHT11 Humidity and Temperature Digital Sensor, Documentation, 2018 March 2018 available at https://www.microbot.it/documents/mr003-005_datasheet.pdf.
[18] RFC 5652, 6.3. Content-encryption process, September 2009.
[19] A. Antoniadis, N. Sklavos, On the white-box cryptography: design and integration of high performance & lightweight encryption, in: Proceedings of 21th EUROMICRO Conference on Digital System Design, Architectures, Methods, Tools (DSD'18), Prague, Czech Republic, 2018 August 29–31.

[20] N. Sklavos, In the era of cybersecurity: cryptographic hardware and embedded systems, 8th Mediterranean Conference on Embedded Computing (MECO'19), 2019 June 10–14.
[21] G.iovanni Danese, M.auro Giachero, F.rancesco Leporati, N.elson Nazzicari, An embedded multi-core biometric identification system, in: Embedded Hardware Design: Microprocessors and Microsystems, Elsevier, 2011, pp. 510–521. Vol. 35, Issue 5.
[22] E.manuele Torti, A.lessandro Fontanella, G.iordana Florimbi, F.rancesco Leporati, H.imar Fabelo, S.amuel Ortega, G.ustavo M.arrero Callicó, Acceleration of brain cancer detection algorithms during surgery procedures using GPUs, in: Embedded Hardware Design: Microprocessors and Microsystems, 61, Elsevier, 2018, pp. 171–178. Vol.
[23] G. Florimbi, E. Torti, S. Masoli, E. D' Angelo, G. Danese, F. Leporati, The human brain project: parallel technologies for biologically accurate simulation of Granule cells, in: Embedded Hardware Design: Microprocessors and Microsystems, 47, Elsevier, 2016, pp. 303–313. Vol.

Paris Panagiotou received his diploma from Computer Engineering & Informatics Department (CEID) at University of Patras, Hellas, (2018). Currently, he is a M.Sc. student of Embedded Systems, at Delft University of Technology (TU Delft), Netherlands. His -research of interests is focused on Security, Embedded Systems, Microprocessors, and Internet of Things (IoT). During his bachelor thesis, he has published a research paper in the field of IoT, which is presented in the DSD Euromicro Conference on Digital System Design (08/2018).

Dr. Nicolas Sklavos, is Associate Professor, with Computer Engineering & Informatics Department (CEID), Polytechnic School, University of Patras, Hellas. He is Director of SCYTALE Group. His research interests include Cryptographic Engineering, Hardware Security, Cyber Security, Digital Systems Design, and IoT. He has participated to a number of European & National, Research and Development Projects. He is Evaluator/Reviewer of project calls, funded by the European Commission, or National Resources. He has participated to the organization of international scientific conferences, of IEEE/ACM/IFIP, serving several committee duties, as well as Editorial Board Member of Scientific Journals. He has authored or co-authored technical papers, books, chapters, reports etc, in the areas of his research. His published works has been cited in several papers of other authors, in technical and scientific literature. He is Senior Member of IEEE and Associated Member of HiPEAC.

Eleni Darra was born in Athens, Greece. She received an M.Sc. in Network Oriented Systems (2008) and a B.Sc. in the field of Digital Systems (2005). Currently she is a Research Associate at the Center of Security Studies supervised by the Ministry of Public Order and Citizen Protection, Hellas and an active member of the SCYTALE Group of the Computer Engineering and Informatics Department of University of Patras, Hellas. She has been also a Network Information Security Officer at the European Union Agency for Network and Information Security. Her expertise lies in Cyber Security and Privacy for Mobile Communications, with a specialization in Wireless Sensors Networks, Cloud Computing Security and Intrusion Detection Systems. She has authored or co-authored journal publications, book chapters and conference proceedings publications.

Dr. Ioannis D. Zaharakis, is Professor with Electrical and Computer Engineering Department, University of Peloponnese, Hellas. He received his B.Sc. in Mathematics (in 1992) and his Ph.D. (in 1999, in Software Engineering) both from the Dept. of Mathematics, Univ. of Patras, Greece. In 2002, he joined the DAISy Unit at Computer Technology Institute & Press "Diophantus" as a Researcher/R&D Engineer, focusing on the formal specification of component-based ubiquitous computer systems; he is Managing Director of the Research Unit DAISy (http://daisy.cti.gr) since 2009. During 1993–1999, he was researcher in the Educational Software Development Lab, University of Patras, where he was involved in the specification, design and implementation of intelligent tutoring systems. He has participated in several EU-funded and national R&D projects. He has authored two textbooks for the Hellenic Open University and more than 30 papers that have been published in international journals and conferences. He is a member of the Hellenic AI Society and the Hellenic Mathematics Society.