Malaysian Journal of Computing and Applied Mathematics 2020, Vol 3(1): 1-10
©Universiti Sultan Zainal Abidin
(Online)

MALAYSIAN JOURNAL OF COMPUTING AND APPLIED MATHEMATICS

Secure Hybrid Crypto-system AES/RSA on FPGA for Data Communication

M Issad[a], N Anane[b], A M Bellemou[c], B Boudraa[d]

[a,b,c]Centre de Développement des Technologies Avancées, CDTA Baba Hassen, Alger, Algérie
[d]Université des Sciences et de la Technologie Houari Boumediene, Bab Ezzouar, Alger, Algerie

[*]Corresponding author: missad@cdta.dz

Received: 21/01/2020, Accepted: 14/04/2020
http://dx.doi.org/10.37231/myjcam.2020.3.1.38

Abstract

With the development of information technologies, our environment is surrounded by digital data that transit via networks. When data are important, they become vulnerable to external attacks which can be avoided by using cryptography which provides confidentiality, integrity and availability required to secure digital data transactions such as e- commerce, mobile telephony and Internet. This paper deals with securing data transmitted over a network composed by a server and several clients, where a security platform has been integrated into the server and embedded on an FPGA circuit. The protection of data transfer between clients is provided by hybrid cryptography combining symmetric and asymmetric cryptographies. The security of client-server communication is ensured by the AES protocol and the Diffie-Hellman key exchange protocol. To offer a good management of keys and their sharing, a dedicated system for generating keys is designed to fit with public key infrastructures. This system is a part of the server and has been implemented using JAVA language and executed on a computer. This communication system provides a Graphical User Interface (GUI) offering to clients ease and flexibility in transferring their data.

Keywords: Diffie-Hellman, Hybrid cryptosystem, AES/RSA, Secure transmission, Virtex-5, FPGA.

INTRODUCTION

Today, more and more sensitive data is stored digitally. Bank accounts, medical records and personal emails are some categories that data must keep secure; this has made cryptography an important research topic.

Cryptography is the science of using mathematics to encrypt and decrypt data. It enables to store sensitive information or transmit it across insecure networks so that it cannot be read by anyone except the intended recipient. Cryptography is the process that involves encryption and decryption of text using various algorithms based on mathematical functions. These algorithms of encryption/decryption can be categorized into symmetric and asymmetric ones.

In symmetric cryptography, the same key is used by the sender and the receiver and must keep it secret. Asymmetric cryptography requires a pair of keys of large size between each two communicators and poses the problem of key distribution. Current cryptographic systems exploit the strengths of both symmetric and asymmetric cryptographies. Symmetric encryption is preferred when confidentiality is required because it is faster as it uses smaller keys than asymmetric encryption.

To provide security and performance while transmitting data via a network, a hardware/software co-design is a good solution to ensure faster speed, more security and consumes less area and power. The hardware implementation on FPGA, which is reconfigurable, offers more flexibility and requires less time to market. This paper focuses on securing transmitted data via a network infrastructure composed by a server and several clients. Figure 1 shows the Client/Server model architecture that is used in most network systems.

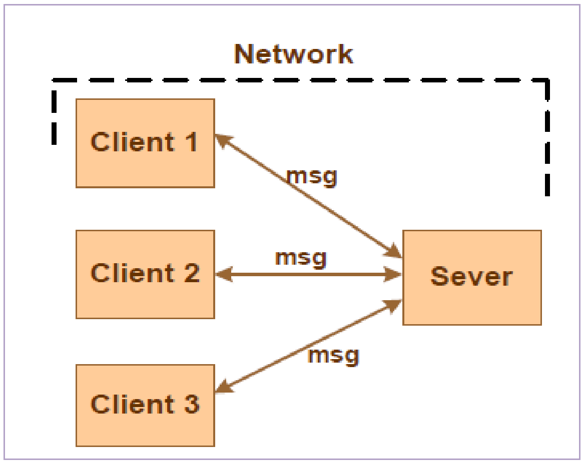

Figure 1. Client/Server architecture

The client side could be any type of smart devices (desktop, laptop, smart phone, etc.). The server part is one device that control and pass messages and opening the connections among clients and/or between clients and server [1]. The Internet part could be one device to isolate the network overall into two main parts: client(s) and server, it could be a switch, a hub, a router or just a cable.

In this paper, we have developed a security platform that has been integrated into the server and embedded on FPGA [2]. The implemented network infrastructure can securely transmit encrypted messages or files via a LAN (Local Area Network) basing on the User Datagram Protocol (UDP) [3], which allows transferring large files across the Internet in real-time with low overhead and less processing. Securing data transfer, between clients, is provided by hybrid cryptography combining symmetric cryptography (AES) and asymmetric cryptography (RSA). The security of the client-server communication is ensured by the AES protocol and the Diffie-Hellman Key Exchange protocol (DHKE) [4].

This paper is organized as follows: Section II elucidates about cryptography, where the Diffie-Hellman protocol, the cryptographic algorithms AES and RSA and their combination in hybrid cryptosystem are detailed. Section III, presents the proposed hardware architecture for secure transmitting data over a network. Results and discussions are given in section IV. Finally, in Section V, a brief conclusion is drawn.

CRYPTOGRAPHIC PROTOCOLS

In the network security system, cryptography plays a vital role for secure transmitting information. Cryptography is a process of integrating and transferring data to users against any attacks. There are two types of cryptographic algorithms: symmetric and asymmetric.

In the symmetric cryptography, shown in Figure 2, one secret key is used for both encryption and decryption. The problem with this method is that you have to communicate the secret key securely to your intended recipient. Symmetric algorithms are fast and simple to implement since they use small keys sizes.

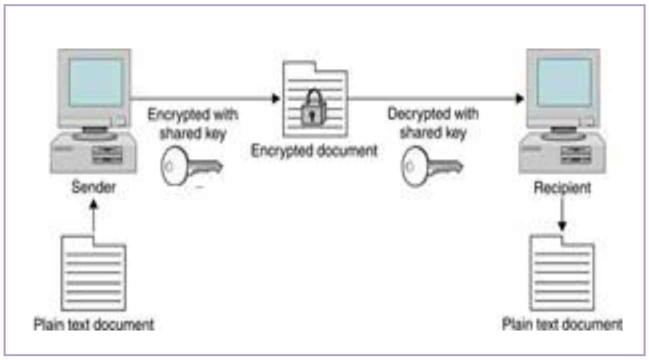

Figure 2. Symmetric Cryptography

Asymmetric cryptography, shown on Figure 3, uses a pair of keys: a public key to encrypt the message at sender and a private key known only to receiver for decrypting the encrypted message. Asymmetric algorithms are more secure but require a huge amount of calculus since they use large keys sizes to encrypt the message [5].

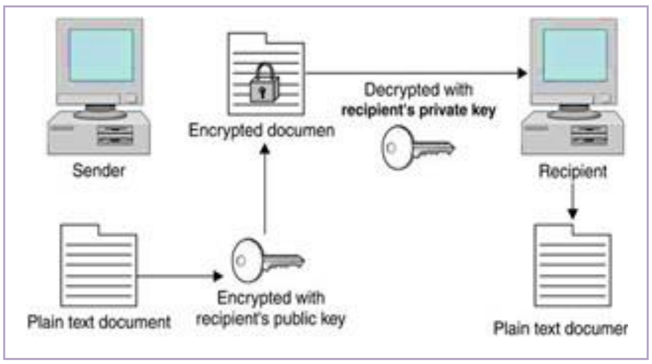

Figure 3. Asymmetric Cryptography

The concept of public key cryptography was introduced by Diffie-Hellman in1976. Their contribution was the notion that keys could come in pairs (encryption and decryption keys) and that one could not generate one key from the other. Since 1976, numerous public key cryptosystems have been proposed and the secure and practical ones are the Diffie Hellman Key exchange (DHKE) and the Rivest Shamir Adleman (RSA).

A. Diffie-Hellman Key Exchange protocol

The DHKE is a secure method for exchanging cryptographic keys over an insecure channel. This method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure channel. This key can then be used to encrypt subsequent communications using a symmetric key cipher. The simplest implementation of the protocol uses the multiplicative group of integers modulo p, where p is a prime and g is a primitive root modulo p. These two values are chosen in this way to ensure that the resulting shared secret can take on any value from 1 to p–1.

Here is an example represented on Figure 4, where A and B agree to use a modulus p and a base g. A chooses a secret integer “a”, then sends to B, $A = g^a \bmod p$, where B chooses a secret integer “b”, then sends to A, $B = g^b \bmod p$.

- A computes $K = B^a \bmod p$
- B computes $K = A^b \bmod p$

A and B now share a secret $K= B^a \bmod p= (g^b \bmod p)^a \bmod p = g^{ab} \bmod p = (g^a \bmod p)^b = A^b \bmod p$.

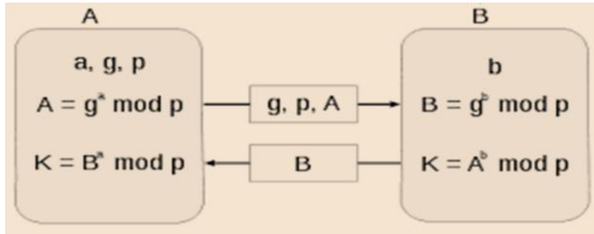

Figure 4. Diffie Hellman key exchange protocol

B. RSA Cryptosystem

The RSA crypto system, shown on Figure 5, was called for their inventors named: Ronald Rivest, Adi Shamir, and Leonard Adleman. It is a public key encryption algorithm developed for signing and encrypting. It is still widely used in electronic commerce protocols, and is believed that its security depends on the difficulty of decomposing large numbers. RSA is secure because it is able to resist concerted attack and is based on modular exponentiation of large sizes integers.

The computational steps for key generation of RSA are described in the following steps [6]:

1. Generate two different primes p and q of the same length.
2. Calculate the modulus $n = p \times q$.
3. Calculate the quotient $\varphi(n) = (p - 1) \times (q - 1)$.
4. Select for public exponent an integer e such that: $1 < e < \varphi(n)$ and $\gcd(\varphi(n), e) = 1$.
5. Calculate for the private exponent a value for d such that: $d = (e^{-1}) \bmod \varphi(n)$.
6. Public Key = {e, n}. Private Key = {d, n}.
7. Encrypting a message m is computing $c = m^e \bmod n$
8. Decrypting the message c is computing $m = c^d \bmod n$.

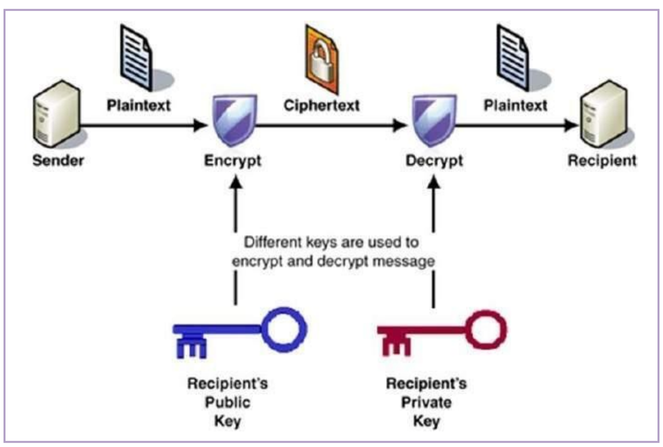

Figure 5. RSA crypto system

C. Advanced Encryption Standard

Advanced Encryption Standard (AES) NIST (2001), is an algorithm used for data encryption. AES is a part of the symmetric block cipher family, which is working with blocks of data, and they are of fixed length (128 bits). These bits are placed to matrix of 4×4, when one cell of matrix corresponds to one byte. One key of length 128, 192 or 256 bits is used for encryption and decryption. In this paper we are working with 128 bits key. This algorithm is shown on Figure 6.

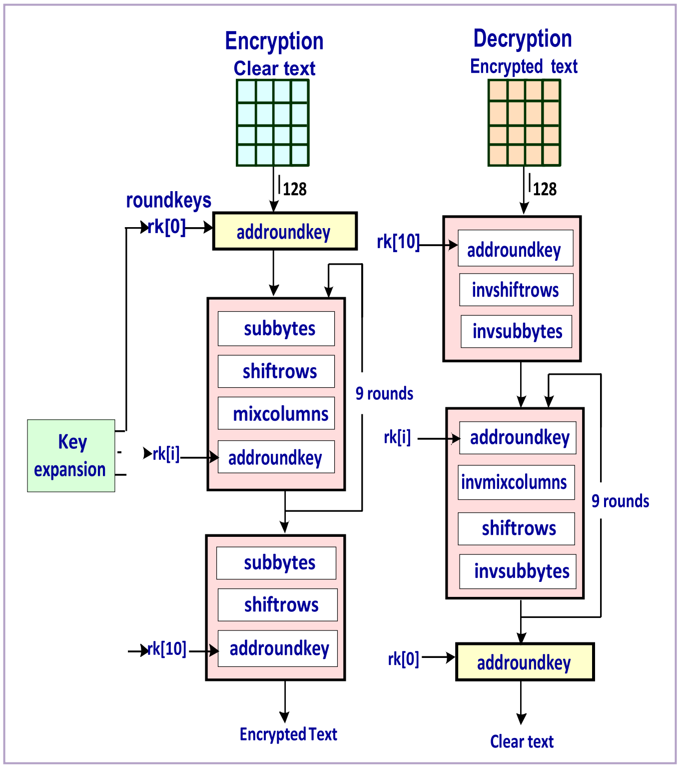

Figure 6. AES Algorithm

The AES cryptosystem can be divided into three parts [7]:

- Initial part (Key Expansion, AddRoundKey),
- Iteration part– so called round (SubBytes, ShiftRows, MixColumns, AddRoundKey),
- Final part (SubBytes, ShiftRows and an AddRoundKey).

An expansion of the key is performed at the beginning of the encryption. In the cipher XOR operation between the 128 bits key and the 4×4 state matrix (block of 128 bits of data) is performed. Subsequently, nine iterations which are normally referred to as the round are performed. The number of rounds depends on the length of the key, for 128 bits key it is 9. Every round consists of the substitution of the bytes in the state matrix (Sub Bytes), rotation of rows (Shift Rows), and substitution of columns (Mix Columns).

The matrix is combined with round's key (Add Round Key), at the end of each round. The final part consists of the substitution of the bytes, rotation of rows and the last addition of the round key. Bytes of the cipher text are stored in the resulting matrix.

D. Hybrid Cryptosystem

Symmetric and asymmetric cryptosystems have their own advantages and disadvantages. Symmetric cryptosystems are significantly faster than asymmetric ones, but require all parties to somehow share a secret (the key). Asymmetric cryptosystems are secure and allow public key infrastructures and key exchange systems, but at the cost of speed since they use big size keys hence require a huge amount of calculus.

A hybrid cryptosystem combines the symmetric and asymmetric cryptographies in order to benefit from the rapidity of one and the security of the other. It offers better efficiency and performance.

The hybrid cryptosystem is shown in Figure 7, which consists in generating a random secret key for a symmetric cipher, and then encrypting this key via an asymmetric cipher using the recipient's public key. The message itself is then encrypted using the symmetric cipher and the secret key. Both the encrypted secret key and the encrypted message are then sent to the recipient.

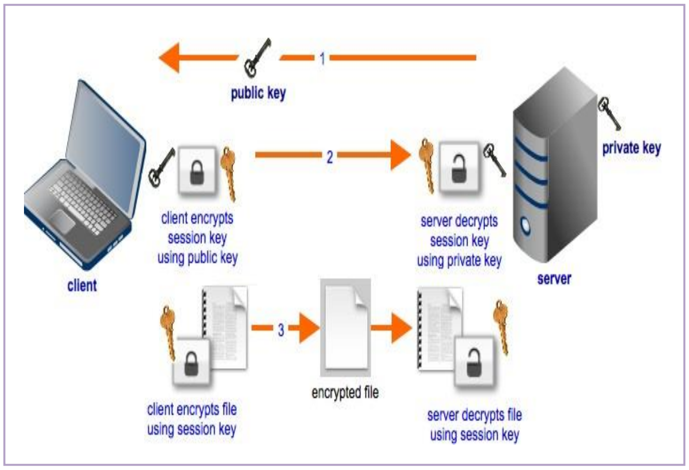

Figure 7. Hybrid cryptography

Hybridization deploys the convenience of the public-key cryptosystem RSA and the efficiency of the private key cryptosystem AES. AES is usually high speed and requires low RAM, so it is very useful for encrypting data, but since it's the same key for both encryption and decryption, there is a big problem of key transport from the encryption side (sender) to the decryption side (receiver). RSA is used to protect the encryption key by generating two keys (private and public). The private key stays on the receiver side and the public key is sent to the sender side. The sender will use this key to encrypt data.

THE PROPOSED ARCHITECTURE

The architecture of our security infrastructure is shown on Figure 8. The proposed platform is dedicated to transmit data securely via a LAN and is implemented on an FPGA circuit. It is based on hybrid cryptographic protocols combining AES, RSA and the Diffie-Hellman key exchange protocol. This design is composed by two parts: the server and clients where their communication is based on the UDP protocol. It is composed of the server and clients linked by an Ethernet communication.

A. The server

It is composed of two sub-parts connected by an RS232 link.

1. An FPGA prototyping card embedding a security platform which roles are:
    a. Computation of the modular exponentiation required by the DHKE protocol for the creation of secure server-client channel.
    b. AES encryption/decryption on the server.
    c. Data emission/reception between clients via an Ethernet link.
    d. Data emission/reception with the IHM server via the RS232 serial link.
2. An RSA key generator and a database, which roles consist in:
    a. Generating random numbers used in the DHKE protocol.
    b. Generating RSA public and private keys associated to each client.
    c. Transmitting Data to the FPGA circuit.
    d. Receiving IDs clients and storing them in the database.
    e. Displaying the database.

B. The client

The client part concerns the IHM which tasks are:

- Generating of a random number used in the DHKE.
- AES encrypting/decrypting during server-client and client- client communications.

- RSA encrypting/decrypting during client-client communication.
- Transmitting/receiving data to the part implemented on the FPGA circuit via the Ethernet link.
- Transmitting/receiving data between clients via Ethernet link and displaying data received in clear.

This IHM offers to users a flexible use with the following options:

- Registration of the client who requests to be added to the database.
- Connection of the client who wants to be connected to the infrastructure.
- Sending a message from one client to another.
- Login out of the client from the infrastructure.

The designed infrastructure ensures secure communications between several communicating clients around the server. This security is provided by either encryption / decryption of data transmitted in real time, or by transmission of files stored in the internal memory of the client computer. The functionality of this infrastructure is structured as shown on Figure 8.

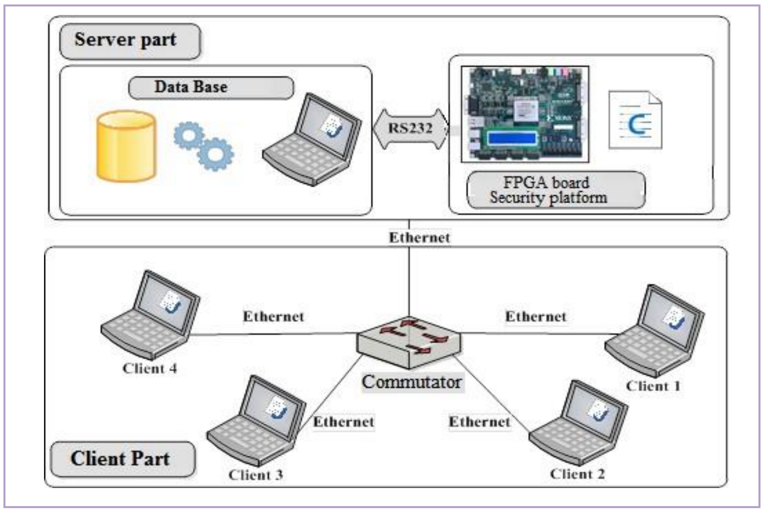

Figure 8. Secure infrastructure architecture

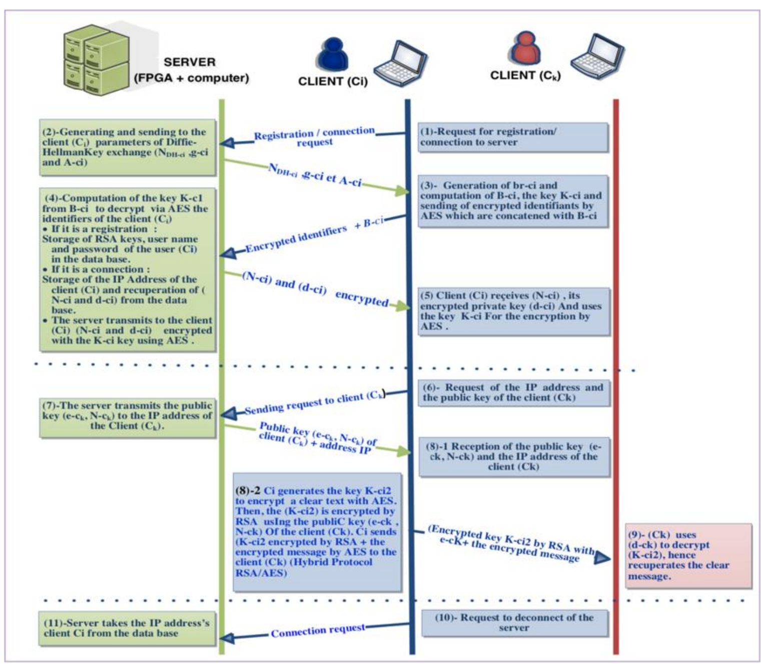

Figure 9. Communications Client-Server and Client-Client

Malaysian Journal of Computing and Applied Mathematics 2020, Vol 3(1): 1-10

(Online)

The communication process using the proposed infrastructure is composed by several tasks accomplished by the server, the client Ci and another client Ck. These tasks are numbered from 1 to 11 and executed in the numbering order as shown on Figure 9.

IMPLEMENTATION RESULTS

To conceive the server-client infrastructure, two tasks have been executed:

1. Hardware implementation of an embedded crypto system by using XPS (Xilinx Platform Studio) and its programming in C language by using SDK (Software Development Kit.
2. Creation of an IHM for the client and a database using JAVA of Netbeans.

As shown in Table 1, the performances of the embedded cryptosystem on the FPGA circuit (the XCVLX50T) concern the execution time of the AES encryption/decryption represented by (tAES), the computation time of a modular exponentiation for a 128-bits exponent represented by (texp) and the occupied hardware resources.

Table 1. Execution performances on FPGA circuit

Execution time			Occupied area		
tAES (encrypt)	tAES (decrypt)	texp	Slices	Memory blocks	DSP blocks
0.19 ms	0.34 ms	8.5 ms	1323	18	3

The developed interface offers to the user the following tasks:

1. Secure clients' connection to the server.
   To connect to the server, the user authenticates by his username and his password which are encrypted by the AES protocol.
   When the server does not recognize the client, it asks him to create a new account as shown on Figure 10.

Figure 10. Authentication window

2. Secure clients registration in the database server.
   The client registers on the server in order to be added to the database using his username and his password. If his username is already stored in the database, the server asks him to change it as shown on Figure 11.

Figure 11. Clients Registration window

3. Secure communication in real time between two clients.
   Once clients are registered or connected, they can communicate by transmitting and receiving information (files) between them as shown on Figures 12 and 13.

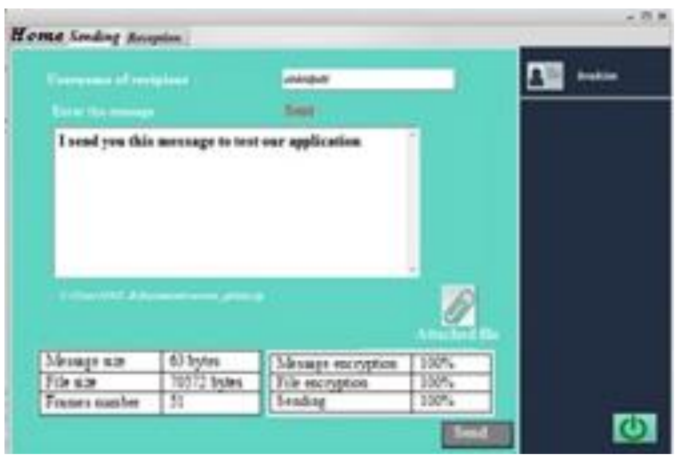

Figure 12. Data sending window

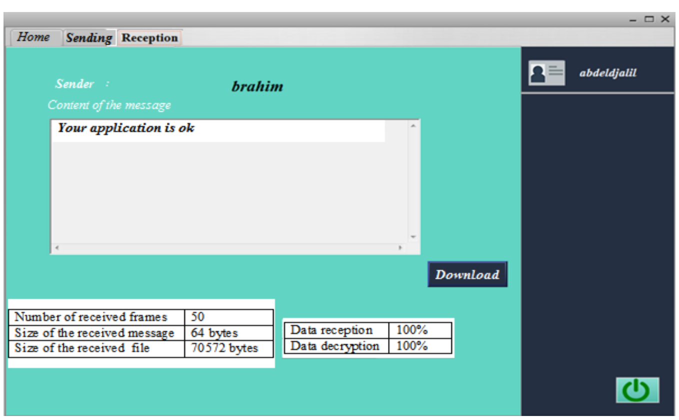

Figure 13. Data reception window

CONCLUSION

This paper has presented a security infrastructure based on hybrid crypto system AES/RSA embedded on FPGA circuit. This application can be used to transmit messages and encrypt files securely over a LAN. Experimental results have shown that the developed security infrastructure exhibits obvious speed and performance advantages in comparison with related works and offers more security. Our security infrastructure can be considered as a first prototype, where all the critical operations are embedded on FPGA circuit. In perspective, this work can be improved by the integration of a random number generator on the part implemented on FPGA circuit.

ACKNOWLEDGEMENT

The authors would like to thank the Directorate General for Scientific Research and Technological Development of Algeria for funding this work through research.

References

Bonde V V, Kale A D. (2015). Design and Implementation of a Random Number Generator on FPGA. International Journal of Science and Research, 4(5), 203-208.

Diffie W and Hellman M. (1976). New Directions in Cryptography, IEEE Transactions on Information Theory, IT-22.

Harba E S I. (2017). Secure Data Encryption Through a Combination of AES, RSA and HMAC. Engineering, Technology & Applied Science Research, 7(4), 1781-1785.

Mantoro T, Zakariya A. (2012). Securing E-Mail Communication Using Hybrid Cryptosystem on Android-based Mobile Devices, TELKOMNIKA, 10(4), 807~814.

Nayak V N, Kumar M R, Anusha K, Kiran C K. (2018). FPGA based asymmetric crypto system design. International Journal of Engineering & Technology, 7(1.1), 612-617.

Smekal D, Frolka J, Hajny J. (2016). Acceleration of AES Encryption Algorithm Using Field Programmable Gate Arrays, IFAC-Papers OnLine 49(25), 384–389.

Sonmez F, Al-Bayati J S H. (2017). Development of a Client / Server Cryptography-Based Secure Messaging System Using RSA Algorithm. Journal of Management Engineering and Information Technology, 4(6), 2-6.

Zodpe H, Sapkal A. (2018). An efficient AES implementation using FPGA with enhanced security features. Journal of King Saud University – Engineering Sciences 32(2), 115-122.