RSA Algorithm and LSB Steganography

G.Suman [1]
Electronics and Communication Engineering
SR Engineering College
Warangal, India

P.Anuradha [2](M.Tech)
Electronics and Communication Engineering
SR Engineering College
Warangal, India.

Abstract: In this paper we are proposing a novel technique for encrypting a message for network security application. Here we are applying both RSA algorithm and LSB steganography method for message to provide higher security. This algorithm was developed using system C coding and implemented on FPGA. FPGA will provide the quantified architecture for development an ASIC IC.

Keywords: RSA, LSB Steganography, FPGA

I. INTRODUCTION

The earliest forms of information hiding can actually be considered to be highly crude forms of private-key cryptography; the "key" in this case being the knowledge of the method being employed (security through obscurity). Steganography books are filled with examples of such methods used throughout history. Greek messengers had messages tattooed into their shave head, concealing the message when their hair finally grew back. Wax tables were scraped down to bare wood were a message was scratched. Once the tablets were re-waxed, the hidden message was secure [15]. Over time these primitive cryptographic techniques improved, increasing speed, capacity and security of the transmitted message.

Today, crypto-graphical techniques have reached a level of sophistication such that properly encrypted communications can be assumed secure well beyond the useful life of the information transmitted. In fact, it's projected that the most powerful algorithms using multi kilobit key lengths could not be comprised through brute force, even if all the computing power Worldwide for the next 20 years was focused on the attack. Of course the possibility exists that vulnerabilities could be found, or computing power breakthroughs could occur, but for most users in most applications, current cryptographic techniques are generally sufficient.

A public-key encryption technology developed by RSA Data Security, Inc. The acronym stands for Rivest, Shamir, and Adelman, the inventors of the technique. The RSA algorithm is based on the fact that there is no efficient way to factor very large numbers. Deducing an RSA key, therefore, requires an extraordinary amount of computer processing power and time.

The RSA algorithm has become the de facto standard for industrial-strength encryption, especially for data sent over the Internet. It is built into many software products, including Netscape Navigator and Microsoft internet Explorer.

The RSA algorithm is the most commonly used encryption and authentication algorithm and is included as part of the Web browsers from Microsoft and Netscape. It's also part of Lotus Notes, Intuit's Quicken, and many other products. The encryption system is owned by RSA Security. The company licenses the algorithm technologies and also sells development kits. The technologies are part of existing or proposed Web, Internet, and computing standards.

A.STEGANOGRAPHY IN HISTORY

Steganography means to hide secret information into innocent data. Digital images are ideal for hiding secret information. An image containing a secret message is called a cover image.

First, the difference of the cover image and the stego image should be visually unnoticeable. The embedding itself should draw no extra attention to the stego image so that no hackers would try to extract the hidden message illegally. Second, the message hiding method should be reliable. It is impossible for someone to extract the hidden message if she/he does not have a special extracting method and a proper secret key. Third, the maximum length of the secret message that can be hidden should be as long as possible.

"Steganography is the art of hiding information in ways that prevent the detection of hidden messages."

B.STEGANOGRAPHY IN THE DIGITAL AGE

Steganography is the art of secret communication. Its purpose is to hide the very presence of communication as opposed to cryptography whose goal is to make communication unintelligible to those who do not possess the right keys. Digital images, videos, sound files, and other computer files that contain perceptually irrelevant or redundant information can be used as "covers" or carriers to hide secret messages. After embedding a secret message into the cover-image, a so-called stego-image is obtained. It is important that the stego-image does not contain any easily detectable artifacts due to message embedding. A third party could use such artifacts as an indication that a secret message is present. Once this message detection can be reliably achieved, the steganographic tool becomes useless.

Obviously, the less information is embedded into the cover-image, the smaller the probability of introducing detectable artifacts by the embedding process. Another important factor is the choice of the cover-image. The selection is at the discretion of the person who sends the message. The sender should avoid using cover-images that would be easy to analyze for presence of secret messages. For example, one should not use computer art, charts, images with large areas of uniform color, images with only a few colors, and images with a unique semantic content, such as fonts. Although computer-generated fractal images may seem as good covers[6] because of their complexity and irregularity, they are generated by strict deterministic rules that may be easily violated by message embedding.

C.CRYPTOGRAPHY VS STEGANOGRAPHY

Cryptography is the science of encrypting data in such a way that nobody can understand the encrypted message, whereas in steganography the existence of data is conceived means its presence cannot be noticed. The information to be hidden is embedded into the cover object which can be text, image, audio or video so that the appearance of cover object doesn't vary even after the information is hidden. Information to be hidden + cover object = stego object.

To add more security the data to be hidden is encrypted with a key before embedding. To extract the hidden information one should have the key. A stego object is one, which looks exactly same as cover object with an hidden information.

D.STEGANOGRAPHY VS WATERMARKING

Watermarking is another branch of steganography it is mainly used to restrict the piracy in digital media In steganography the data to be hidden is not at all related to the cover object, here our main intention is secret communication.

In watermarking the data to be hidden is related to the cover object it is extended data or attribute of the cover object, here our main intention is to stop piracy of digital data. Steganography is a very powerful tool because, as the stated above, it can be very difficult to detect.

II PROPOSED SYSTEM

This paper proposed RSA and LSB Information Hiding algorithm to improve more security. The idea behind the LSB algorithm is to insert the bits of the hidden message into the least significant bits of the pixels. Achieving the purpose of information hiding with the secret bits of information to replace the random noise, using the lowest plane embedding secret information to avoid noise and attacks, making use of redundancy to enhance the sound embedded in the way nature to be addressed. The results showed that the proposed algorithm has a very good hidden invisibility, good security and robustness for a lot of hidden attacks.However, the limitation of capacity has led us to think about an improved approach which can be achieved through hardware implementation systems with the help of a programmable gate array (FPGA) board.

In this process first we generating the Public key and private key for RSA Encryption then we started encryption of RSA using public key then cipher is given as a input to Image then LSB encryption was done then at receiver through LSB

decryption he obtains the cipher then through RSA decryption he obtains the original message.

It is the process of embedding data within the domain of another data, this data can be text, image, audio, or video contents. The embedded watermark can be visible or invisible (hidden in such a way that it cannot be retrieved without knowing the extraction algorithm) to the human eye, specified secret keys are taken into consideration in order to enhance the security of the hidden data

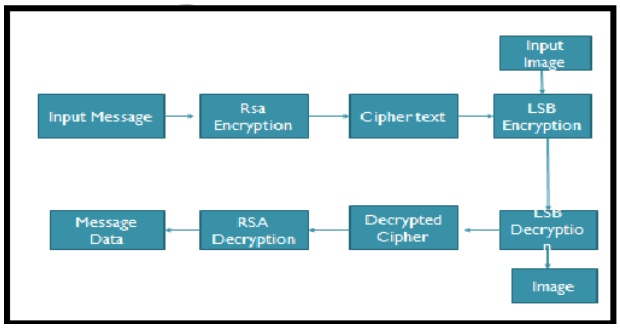

Figure 1: Overall Architecture Block Diagram

III EXPERIMENTAL SETUP

A. Xilinx Platform Studio

The Xilinx Platform Studio (XPS) is the development environment or GUI used for designing the hardware portion of your embedded processor system. B. Embedded Development Kit Xilinx Embedded Development Kit (EDK) is an integrated software tool suite for developing embedded systems with Xilinx Micro Blaze and PowerPC CPUs. EDK includes a variety of tools and applications to assist the designer to develop an embedded system right from the hardware creation to final implementation of the system on an FPGA. System design consists of the creation of the hardware and software components of the embedded processor system and the creation of a verification component is optional. A typical embedded system design project involves: hardware platform creation, hardware platform verification (simulation), software platform creation, software application creation, and software verification. Base System Builder is the wizard that is used to automatically generate a hardware platform according to the user specifications that is defmed by the MHS (Microprocessor Hardware Specification) file. The MHS file defines the system architecture, peripherals and embedded processors]. The Platform Generation tool creates the hardware platform using the MHS file as input. The software platform is defmed by MSS (Microprocessor Software Specification) file which defines driver and library customization parameters for peripherals, processor customization parameters, standard 110 devices, interrupt handler routines, and other software related routines. The MSS file is an input to the Library Generator tool for customization of drivers, libraries and interrupts handlers.

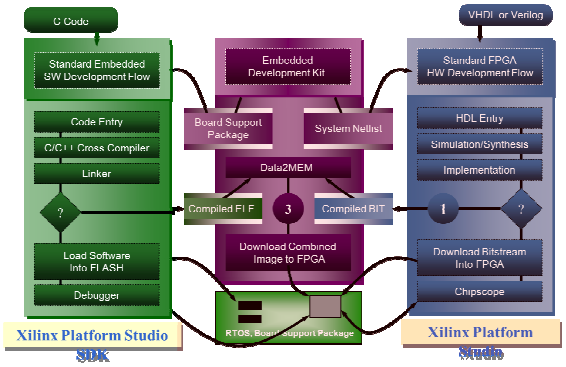

Figure2: Embedded Development Kit Design Flow

The creation of the verification platform is optional and is based on the hardware platform. The MHS file is taken as an input by the Simgen tool to create simulation files for a specific simulator. Three types of simulation models can be generated by the Simgen tool: behavioral, structural and timing models. Some other useful tools available in EDK are Platform Studio which provides the GUI for creating the MHS and MSS files. Create / Import IP Wizard which allows the creation of the designer's own peripheral and import them into EDK projects. Platform Generator customizes and generates the processor system in the form of hardware netlists. Library Generator tool configures libraries, device drivers, file systems and interrupt handlers for embedded processor system. Bit stream Initialize tool initializes the instruction memory of processors on the FPGA. GNU Compiler tools are used for compiling and linking application executables for each processor in the system [6]. There are two options available for debugging the application created using EDK namely: Xilinx Microprocessor Debug (XMD) for debugging the application software using a Microprocessor Debug Module (MDM) in the embedded processor system, and Software Debugger that invokes the software debugger corresponding to the compiler being used for the processor. C. Software Development Kit Xilinx Platform Studio Software Development Kit (SDK) is an integrated development environment, complimentary to XPS, that is used for C/C++

embedded software application creation and verification. SDK is built on the Eclipse open source framework. Soft Development Kit (SDK) is a suite of tools that enables you to design a software application for selected Soft IP Cores in the Xilinx Embedded Development Kit (EDK).The software application can be written in a "C or C++" then the complete embedded processor system for user application will be completed, else debug & download the bit file into FPGA. Then FPGA behaves like processor implemented on it in a Xilinx Field Programmable Gate Array (FPGA) device.

IV CONCLUSION

In this paper we have presented a new method of implementing both cryptography and steganography with higher embedding capacity. Theembedding capacity of the approach is controlled through the filter cut-off frequency. The approach was analyzed and shown to have a very high confidentiality due to the sharpness of information recovery with the cut-off frequency. The New Approach is using lesser hardware architecture with an reasonable speed of 83MHz Approximately in a Spartan 3 FPGA with 50MHz Clock Crystal. So we can use it to the steganography technique very easily than other techniques without any problem.

REFERENCES

[1]. B. Weaver, Now You See It, Scientific Computing 24.6 (May 2007): 18-39.
[2]. B. Glass, Hide in Plain Sight, PC Magazine 21.18 (15 Oct. 2002): 75.
[3]. Tucker, Patrick. "Hiding Secrets in Computer Files." Futurist 40.5 (Sep.2006): 12-12.
[4]. R. Gonzales, and R. Woods, Digital Image Processing, Addison Wesley Publishing Co., 1993.
[5]. C. Birslawn, Fingerprint Go Digital, Notices of American Mathematical Society, Vol. 42, No.11, P. 1278-1283, Nov. 1995.
[6]. W. Sweldens, Building Your Own Wavelets at Home, Wavelets in Computer Graphics, ACM SIGRAPH Course Notes, 1996.
[7]. A. Calderbank, I. Daubechies, W. Sweldens, and B. Yeo, Wavelet Transforms that Map Integers to Integers, Mathematics Subject Classification, 42C15, 94A29, 1996.
[8]. Xilinx, http://www.xilinx.comlproducts/design.
[9]. Xilinx Inc., PicoBlaze 8-bit Embedded Microcontroller UserGuide. http://www.xilinx.comlsupport/documentation/ userJluides/ugl29.
[10]. Digilent Inc., Digilent Nexys2 Board Reference Manual
[11]. Xilinx. Inc., Platform Specification Format Reference Manual, Embedded Development Kit EDK 9.2i
[12]. Xilinx Inc. MicroBlaze Reference Manual, version 10.1.
[13]. Xilinx Inc. Xilinx ISE and Xilinx EDK tools.
[14]. Spartan-3 Starter Kit Board User Guide, Xilinx, Inc.
[15]. Embedded System Tools Reference Manual, Xilinx, Inc
[16]. Sparta n-3 FPGA Family: Complete Data Sheet
[17]. Platform Studio User Guide, Xilinx, Inc.
[18]. Xilinx. Inc., Platform Specification Format Reference Manual, Embedded Development Kit EDK 9.2i
[19]. Xilinx, Embedded System Example, XAPP433, version 2.2, 2006.
[20]. Forchheimer R. (1999), Image coding and data compression, Linköping:Department of electrical engineering at Linköpings University.
[21]. Chui C. K. (1992), An introduction to wavelets, Boston, Academic Press, ISBN 0121745848