Journal of Theoretical and Applied Information Technology
20th April 2013. Vol. 50 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE DESIGN AND IMPLEMENTATION OF SOFTWARE REGISTRATION MODULE BASED ON RSA ENCRYPTION ALGORITHM

[1]JIQIU DENG, [2]BO BI, [3]QIANHONG WU, [4]NA LI

[1,2,3,4]Key Laboratory of Metallogenic Prediction of Nonferrous Metals, Ministry of Education, Changsha, 410083, China

[1,2,3,4]School of Geosciences and Info-Physics, Central South University, Changsha, 410083, China

E-mail: [1]djq318@163.com, [2]whatisbibo@163.com, [3]qhwu19@163.com, [4]anlysmile@qq.com

ABSTRACT

Based upon RSA encryption algorithm and MD5 hash function, this paper designs a software license scheme which adopts the idea of "one machine one code". Using c# language in the .NET Framework 4.0 platform, the author completed the license scheme and it ran on mining management information system successfully. It not only ensures the safety of the software and convenience in registration authorization, but also makes registration code more aesthetic and effective.

Keywords: *RSA Encryption Algorithm, Software License, MD5*

1. INTRODUCTION

For professional software products, software developers must develop software registration authorization module in order to prevent illegal software theft and transfer. Currently, the main software encryption methods are symmetric cryptosystem and asymmetric cryptosystem. Symmetric cryptosystem mainly covers AES (Advanced Encryption Standard), DES (Data Encryption Standard), IDEA (International Data Encryption Algorithm) and so on. Symmetric cryptosystem security depends on the confidentiality of secret key, so security is not enough. The main stream of asymmetric cryptosystem which is also called public key cryptosystem is divided into three parts: based on the difficulty of the big integer factorization, the difficulty of the discrete logarithm calculation and elliptic curve public key cryptography.

The RSA encryption algorithm was first proposed by Rivest RL, Shamir A, Adleman L in 1978. Many scholars have proposed the methods to generate its keys (Boneh, D., Franklin, M.K. 2001; E. Fujisaki and T. Okamoto 1999; Marc Joye 2001). Ming-Der Shieh(2008) presented a new modular exponentiation architecture with a unified modular multiplication/square module which speeded up computation. E. Fujisaki, T. Okamoto, D. Pointcheval, J. Stern(2004) proved the security of RSA-OAEP.

In terms of software Registration and authorization, Xu Su-juan(2006) put forward a software registration program in which Xu Su-juan encrypts the hard disk serial number, MAC address with MD5 encryption method as the registration number and encrypts the registration number with RSA encryption method for storage. As the registration number was encrypted with MD5 method, the software security was not high but the registration number was saved safely. Zhao LiPing, Shu QiLiang, Lai XiaoLiang(2011) proposed a file encryption scheme based on RSA encryption algorithm and determined whether the files had been tampered with MD5 Algorithm. The encryption and decryption method had been implemented of a single text. Liu Wentao(2012) summarized the software license protection program and proposed that not only one of the software protection technologies should be applied. Multiple encryption and verification technologies should be combined. A method that used the exe file and DLL file to protect each other and used the license file to check and provide software function was put forward. Tang Jiutao, Lin Guoyuan divided the forms of software protection into software-based encryption and hardware-based encryption. The two methods are usually used together with no

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

clear boundaries. Tang Jiutao, Lin Guoyuan(2010) discussed the data encryption technology, antidisassembly, antidebugging, code obfuscation, software watermarking, and proposed two trends of the software protection technology. Zhu Chuang-lu(2012) put forward a software protection scheme based on the Internet and ensured the software product the only authorized copy characteristics though charac-teristic vector. In order to prevent skip registration and changing the software, Li Zhi-wei(2012) proposed a method which embedded the charac-teristics of the computer hardware infor-mation into the program internally and protected the software through authentication. The method combined the software encryption with the hardware encryption, but the encryption and decryption algorithms had not been given. Trishna Panse, V. Kapoor.(2012) put forward a scheme that provided the integration of Triple DES, RSA and MD5 in Bluetooth communication.

Due to the RSA encryption algorithm based on big integer factorization, it is a NP-complete problem and there is no effective method to crack currently. Note that the RSA encryption algorithm used for software registration can protect software copyright effectively. However, only use the RSA encryption method to generate registration code cannot ensure its length and it is inconvenient to the registration code storage management and the registration authorization. In addition, the regis-tration code is not aesthetic as well. According to the present situation, the author puts forward the following requirements: the registration code should generate in the traditional software regis-tration authorization mode; the verification process should be safe, the registration code should be beautiful and convenient to input. Therefore, this paper uses the "one machine one code" registration model to design and implement the software registration module based on RSA encryption algorithm in the .NET Framework 4.0. In fact, the module has been used in mining information management system and it can make the software registration safe, aesthetic, simple, convenient and effective.

In section 2, we propose the idea of generating registration code and the module design. In section 3, we provide the implementation of the module program. Section 4 presents the results and analysis. Section 5 gives a conclusion to the whole paper.

2. ALGORITHM IDEA AND MODULE DESIGN

2.1 Generate Registration Code

In consideration of the uniqueness of CPU serial number in computers, CPU serial number can be served as machine code. The user who needs software authorization should provide machine code to software provider, then software providers use RSA encryption algorithm to encrypt machine code and transform the encrypted results into the MD5 hash value. Because results have a fixed length after MD5 encryption, aesthetic and fixed length registration code can be generated through proper handling. The Registration code will be sent to the user so that the user can use the software through the registration verification. Encryption process is as follows:

In RSA encryption algorithm, it needs to find large prime numbers p and q, calculate n =pq, take a > 1 meeting $(a,\varphi(n))=1$, then find d meeting $da \equiv 1(\mathrm{mod}\varphi(n))$. Encryption is obtained by $Ea \equiv m^a(\mathrm{mod}\ n)$ while decryption is obtained by $Ea^d \equiv m(\mathrm{mod}\ n)$. Here p, q, d, $\varphi(n)$ are all secret keys and should be remain private. In the software registration module, read the user's CPU serial number first. With a set of established secret keys, encrypt the machine code to get an encryption result Ea. And the code with an arbitrary length can be operated into a fixed length value by MD5. Then turn the result Ea into MD5 hash value recorded as EEa, remove any four position value in EEa and insert "-" in the position NO.15, 10, 5 to get the final registration code. The specific implementation steps are as follows:

1) Generate large prime number p, q. Big prime number can be generated through the Maple. In Maple, through the commands such as "ithprime", "nextprime" and "prevprime", prime number can be generated easily. For example, through the Function (1) a large prime number more than 100 bit can be generated;

$$p := \mathrm{nextprime}(123!); \quad (1)$$

2) Calculate $n = p \cdot q,\ t = (p-1)\cdot(q-1)$;

3) Select a > 1, such as take 67, to calculate t mod 67, if not zero, then a = 67;

4) Calculate $d := \frac{1}{a} \bmod t;$ so to get a group secret keys;

5) In order to use c# depict secret key conveniently, first the key should be

Journal of Theoretical and Applied Information Technology
20th April 2013. Vol. 50 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

convert into hexadecimal form. In the system, with byte[] type initialization, .NET 4.0 provides the BigInteger class to depict any big signed integer [2] and byte[] type can be forced into BigInteger type;

6) Read the CPU serial number and turn it into byte sequence (byte[]), then convert to BigInteger type as data m which needs to encrypt;

7) Use the selected secret key to encrypt m, then get Ea;

8) Convert Ea to MD5 hash value EEa;

9) Remove any four position values in EEa, and insert "-" in its 15, 10, 5 locations to get the final registration code.

2.2 The Design of the Software Registration Module

Taking into account the system's ease of maintenance and scalability, let the software registration module be a project alone in Visual Studio 2010 programming environment. A class that inherits from the Installer is added to the project. Override the Install method and a form will appear during its execution. Users must enter a registration number to continue. The authentication of the registration number will execute in the form. So the authentication and authorization can be performed during the software installation process.

The registration number must be entered during the software installation process. The software cannot be installed, unless you enter the correct registration number. After successful validation, registration number will be written in a configuration file in the software installation directory. The registration number in the configuration file will be checked and verified before the system runs in order to prevent users from skipping the installing and copying the software to use directly. The machine code will be given and the software supplier has the program to turn the machine code into the registration number. It is convenient and effective to provide authorization and protect the copyright of the software.

3. HE DESIGN AND IMPLEMENTATION OF THE MODULE PROGRAM

Based on .NET Framework 4.0 the software registration program is realized by C# programming language in the Visual Studio 2010 development environment. It can be used for authentication of the registration number. Using the same method the program of generating registration number is written in another solution.

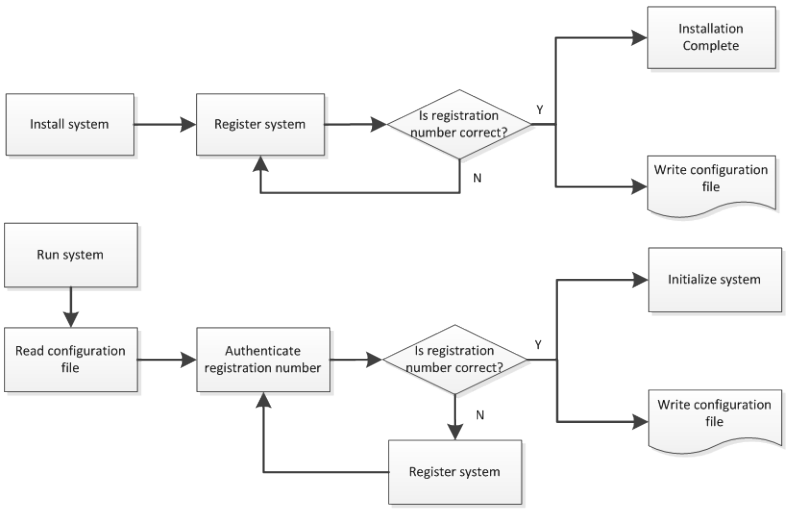

Fig. 1: Install or run the system

3.1 Machine code obtainment

In .NET Framework, the System.Management namespace provides access to a rich set of management information about the system, devices and applications. By using the namespace, it's easy to get the CPU serial number, which can be treated as the machine code.

Journal of Theoretical and Applied Information Technology
20th April 2013. Vol. 50 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.2 Realization of Encryption

Public and private keys can be designed up front on Maple and initialized in the program by C#. Using ModPow() method of the class BigInteger can realize encryption. The result returned is a BigInteger and it can be transformed into a byte[] by the method ToByteArray(). So the method ComputeHash of the class MD5CryptoService-Provider can be used to encrypt the result to its MD5 hash value. Finally, appropriately adjust the MD5 hash value to the formation of the "*****-*****-*****-*****-*****" form. It is the ultimate registration number. During the registration and authentication, compare the registration number with the correct one to determine whether the software can be registered or has been registered.

In the key initialization, p and q are all more than one hundred digits large prime numbers. Initialize the hexadecimal numbers to byte[] type and then convert them into BigInteger type. Calculate their product n. a should also be Initialized to the Big-Integer type.

For example data is the machine code that will be encrypted. Encode data into a sequence of bytes and then convert it into BigInteger type. Encrypt it and get the value after encryption. The crucial code of the encryption is as follows:

```
byte[] Endata_byte;

BigInteger rsa_data =new BigInteger(data);

BigInteger Endata_bigint = BigInteger.Mod-Pow(rsa_a,rsa_data,rsa_n);

Endata_byte = Endata_bigint.ToByteArray();
```

Encrypt the result to its MD5 hash value and then turn it into the form of ultimate registration. encpubyte is the result of the RSA encryption and regstr is the ultimate registration number. The crucial code is as follows:

```
MD5CryptoServiceProvider MD5CSP1 = new MD5CryptoServiceProvider();

byte[] registBytestr = MD5CSP1.Compute-Hash(encpubyte);

StringBuilder md5str = new StringBuilder();

md5str.Append(BitConverter.ToString(registB ytestr)).Replace("-", "");

md5str.Remove(6, 3).Remove(8, 3).Remove-(13, 3).Remove(20, 3);

regstr = md5str.Insert(15, "-").Insert(10, "-").Insert(5, "-").ToString().ToUpper();
```

4. RESULTS AND ANALYSIS

The encryption method is tested on about 50 computers including single- and dual-core CPU computer, and the computers with different operating systems including Windows xp, Windows 7, Windows Server 2003. The result conforms to the idea of “one machine one code”.

For example, we tested a computer with Dual-core CPU and Windows7 operating system. Its CPU serial number was “BFEBFBFF0001067A” and its registration number should be “C3A0E-5A3DB-553C7-17C64” (Fig.2).

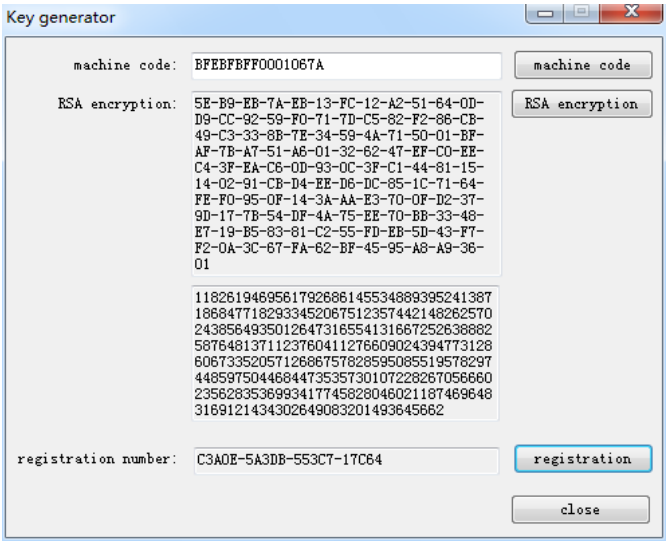

Fig. 2: Acquire machine code and generate registration number

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The computer's CPU serial number is unique and for a given machine code, there's only one registration number. The generated registration numbers are aesthetic, safe and stable. This registration method has been used in a mine information management system. A few computers have been registered without any safety problem. It is a little slow to read the CPU serial number. The serial number of other hardware devices can also be treated as the machine code. If other length of the registration number is needed, the result of the RSA encryption can be dealt with in other ways and generate the registration numbers to meet the demand.

5. CONCLUSION

Based upon RSA encryption algorithm and MD5 hash function, this paper designs a software license scheme which adopts the idea of "one machine one code". Using c# language in the.net Framework 4.0 platform, the author completed the license scheme and ran on mining management information system successfully. It not only ensures the safety of the software and convenience in registration authorization, but also makes registration code more aesthetic and effective.

REFRENCES:

[1] Boneh, D., Franklin, M.K, "Efficient generation of shared RSA keys", *J. ACM* 48(4), 2011, pp.702–722.

[2] MSDN Library – Biginteger. [Online] http://msdn.microsoft.com/zh-cn/library/system.numerics.biginteger%28v=VS.95%29.aspx

[3] E. Fujisaki and T. Okamoto, "How to enhance the security of public-key encryption at minimum cost". In H. Imai and Y. Zheng, editors, PKC'99: *2nd International Workshop on Theory and Practice in Public Key Cryptography, Lecture Notes in Computer Science, Springer* Vol.1560, Mar 1999, pp. 53–68.

[4] E. Fujisaki, T. Okamoto, D. Pointcheval, J. Stern, "RSA-OAEP is secure under the RSA assumption" *Journal of Cryptology*, Vol.17 No.2, 2004, pp. 81–104.

[5] Trishna Panse, V. Kapoor, "An Integrated Scheme based on Triple DES, RSA and MD5 to Enhance the Security in Bluetooth Communication". *International Journal of Computer Applications (0975 – 8887)* Vol.50, No.7, July 2012, pp.45-50.

[6] Rivest RL, Shamir A, Adleman L, "A Method for Obtaining Digital Signatures and Public-key Cryptosystem". *Communications of the ACM*, Vol.21, No.2, 1978 pp:120–126.

[7] Marc Joye. "How to Choose Secret Parameters for RSA-Type Cryptosystems over Elliptic Curves". *Designs, Codes and Cryptography*, 23, 2001, pp.297-316.

[8] Ming-Der Shieh, Jun-Hong Chen, Hao-Hsuan Wu, and Wen-Ching Lin. "A New Modular Exponentiation Architecture for Efficient Design of RSA Cryptosystem". *IEEE Transactions on Very Large Scale Integration (VLSI) System*, Vol.16, No.9, September 2008, pp.1151-1161.

[9] Xu Su-juan. "Based on MD5/RSA algorithm to improve the protection of computer software registration code". *Fujian Computer*. 8(2006) pp.109, 107.

[10] Zhao Li Ping, Shu Qi Liang, Lai Xiao Liang, "RSA Encryption and Digital Signature", *International Conference on Computational and Information Sci*ences.2011.245, pp.369-372.

[11] Wentao Liu. "Software Protection with Encryption and Verification". *Software Engineering and Knowledge Engineering*: Vol. 2, AISC 115, 2012, pp. 131–138.

[12] Tang Jiutao, Lin Guoyuan. "Research of Software Protection. *2010 International Conference on Educational and Network Technology*, pp.410-413.

[13] Zhu Chuanglu. "Research on Software Registration Protection Based on Characteristic Vector". *Computer & Digital Engineering*. Vol.40, No.11, 2012 pp.132-134.

[14] Li Zhi-wei. "Hardware feature information embedded certification based on software protection". *Computer Engineering and Design*. Vol.33 No.7, July 2012, pp.2550-2554.