Eng. & Tech. Journal ,Vol.28, No.1,2010

Design and Implementation of A Software Protection System Against Software Piracy By Using Cryptographic Techniques

Dr. Alia Karim Abdul Hassan* **&** **Zainab M. Fadhel***

Received on: 4/1/2009
Accepted on:3/9/2009

Abstract

Software Piracy becomes a major problem with the fast and vast growth in the use of the internet, and the new computer technologies is aided in increasing software piracy. In this work, a software protection system against piracy is proposed. This proposed system uses standard techniques to ease these difficulties such as Zero knowledge proof, Improved RSA, MD5, and Triple DES. The proposed system use a proposed method to protect all the system files, and a proposed algorithm to generate software Copy Identification Number called (ICN). The implemented system where a software application hashes hardware serial numbers to generate a unique Installation ID. This Installation ID is sent to the manufacturer to verify the authenticity of the application and to ensure that the product is not being used for multiple installations.

Keywords: Software Piracy, Software Piracy, Zero knowledge proof, Improved RSA, MD5, and Triple DES, Copy Identification Number.

تصميم وتنفيذ نظام حماية ضد قرصنة البرمجيات باستخدام تقنيات التشفير

الخلاصة

أصبحت قرصنه البرامج مشكله رئيسيه مع النمو السريع والواسع في استخدام الانترنت ومع التقدم التكنولوجي الذي ساعد في زيادة مشكله القرصنة . في هذا العمل نظام مقترح لحماية البرامج ضد القرصنة .هذا النظام المقترح نظام متكامل يعتمد على المواصفات القياسية مثل Zero knowledge--proof, RSA, MD5, Triple DES لحل هذه المشكلة. نظام الحماية المقترح يحل أكثر أنواع مشاكل القرصنة شيوعا ويكون أمنا ضد التهديدات الهجمات المعروفة .النظام المقترح يستخدم طريقه مقترحه لحماية فايلات النظام بأكملها وخوارزميه مقترحه لإعطاء رقم تعريفي لنسخه البرنامج. النظام المقترح يطور نظام يقوم بقراءة معلومات الأجهزة المتواجدة داخل الحاسوب، ومن ثم بتشفيرها. وبعد ذلك يتم التأكد من أن النسخة المستخدمة من قبل المستخدم لم يتم استخدامها من قبل مستخدمين آخرين، وذلك من خلال إرسال المعلومات إلى المصنع.

https://doi.org/10.30684/etj.28.1.11
2412-0758/University of Technology-Iraq, Baghdad, Iraq
This is an open access article under the CC BY 4.0 license http://creativecommons.org/licenses/by/4.0

Introduction

Software Piracy can be defined as the unauthorized use of commercial software product. It covers any product for which the software developer is not justly rewarded, and is understandably a major concern in the computer market. While the threat of legal action is effective against possible corporate pirates, it is ineffective against the individuals who pirate personal computer softwar[1]. The main objective of all the protection schemes is to raise the cost for pirates to break the protection approaches. Thus, the higher the cost for the pirates to break the software security, the higher the protection level of the application [2]. Software is an intellectual property. It should be protected from unauthorized users in order to ensure that the existing revenue runs. Software piracy continues to grow globally because it is cheap and easy to copy. The effects of this grew are devastating: not only does software piracy reduce revenues, it also results in less research and development, and in less investment in marketing and channel development [3]. In this work a proposed method for software protection that treat most common of software piracy types and this method method based on using cryptographic techniques.

Software Piracy Types

The Business Software Alliance [BSA] defines five common types of software piracy [4]:

1)End-user piracy occurs when a user reproduces copies of software without authorization. It can manifest itself in one of the following forms: a)A user obtains a single licensed copy and uses it to install the software on multiple computers. b)The disks used to install the software are duplicated and then distributed. c)A user purchases and installs an upgrade without previously having a legal version. d)Within a commercial environment, employees use software with an academic license.

2)Client-server piracy occurs when a program is installed on a network and is simultaneously used by more people than the licensed entitled.

3)Internet piracy occurs when illegal copies of software are made available on the Internet either free of charge or for a fee. Examples of such sites include: a)Sites which make software available for free or by exchanging uploaded programs. b)Auction sites that offer illegal software. c)Peer-to-peer network sites which enable the transfer of illegal software.

4)Hard-disk loading occurs when illegal software is installed on a new computer and sold. This activity often occurs when a business is trying to cut costs to make their products more attractive.

5)Software piracy occurs when copyrighted material is illegally duplicated and sold with the intent that the material passes as the original.

Software Protection Methods

Most common techniques for software protection against piracy are classified to Hardware-Based protection, Software-based protection.

Hardware-Based Protection

In this scheme, a tamper-proof, non software-based component is used to authenticate the running software.

The customer attaches this dongle to his system through its external interface. The dongle approach cannot be considered user-friendly or end-user transparent. Flexibility is not one of its strongest points either, every customer must be supplied with the mandatory hardware component, which adds extra costs to both shipment and production. Therefore, dongles are almost exclusively used to protect expensive, professional software packages[5].

Software-Based Protection

Software-based protection techniques are dependant on the same distributed software. Having the software itself as protection model has many advantages such as increasing the distribution flexibility and reducing the protection added cost. One of the most common approaches is Encryption method described in the next section[6].

Encryption

One approach for protecting software is to use encryption. The idea here is to have the distributed software encrypted and a decryption key is needed to execute the software. Many encryption techniques can be used, such as having multiple encryption keys [2]. These techniques for content protection rely on cryptographic techniques in which the decryption key should remain hidden to (illegitimate) users. The flexibility of this scheme allows vendors to limit the time during which the user can enjoy the copyrighted content by only supplying the player with the file's decryption key if the request is legitimate. However, for this feature a user requires an internet connection [7]. In this, work a proposed software protection method using cryptographic techniques. Next chapter review main concept of cryptography with related algorithms and methods used in proposed system.

Cryptography:

Cryptography, simply defined, is the process of combining some input data, called the plaintext, with a user-specified key to generate an encrypted output, called ciphertext, in such a way that, given the ciphertext, no one can recover the original plaintext without the encryption key in a reasonable amount of time. The algorithms that combine the keys and plaintext are called ciphers [8].

Symmetric Cryptography System:

A single secret key which is used in conventional symmetric encryption is used to encrypt and decrypt a message. The most widely used symmetric cipher: is the Data Encryption Standard (DES).although it is destined to be replaced by Advance Encryption Standard (AES), DES remains the most important such algorithm.(though DES's designation has been withdrawn).Despite its withdrawal, DES(especially Triple-DES) remains extremely popular [9]. In this work, TDES is considered and the next section describes it.

Triple DES (TDES)

In cryptography, Triple DES is a block cipher formed from the Data Encryption Standard (DES) cipher by using it three times. When it was found that a 56-bit key of DES is not enough to guard against brute force attacks, TDES was chosen as a

PDF created with pdfFactory Pro trial version www.pdffactory.com

simple way to enlarge the key space without a need to switch to a new algorithm. The use of three steps is essential to prevent meet-in-the-middle attacks that are effective against double DES encryption. Note that DES is not a group; if it were one; the TDES construction would be equivalent to a single DES operation and no more secure. The simplest variant of TDES operates as follows: DES (k3; DES (k2; DES (k1; M))), where M is the message block to be encrypted and k1, k2, and k3 are DES keys [10]. Triple-DES is just DES with two 56-bit keys applied. Given a plaintext message, the first key is used to DES- encrypt the message. The second key is used to DES-decrypt the encrypted message. (Since the second key is not the right key, this decryption just scrambles the data further.) The twice-scrambled message is then encrypted again with the first key to yield the final cipher text. This three-step procedure is called triple-DES. Triple-DES is just DES done three times with two keys used in a particular order. (Triple-DES can also be done with three separate keys instead of only two. In either case the resultant key space is about 2^{112} [11].

Asymmetric Cryptography System
In a modern branch of cryptography also known as public-key cryptography the algorithms employ a pair of keys (a public key and a private key) and use a different component of the pair for different steps of the algorithm[12]. The public-key algorithm relies on one key for encryption and a different but related key for decryption[13] .The many public key algorithms used and the most popular one is RSA .In this work an improved RSA is considered.

Improved RSA Public Key Encryption Scheme[14] The RSA scheme is a block cipher in which the original message and cipher message are integer values in the interval $[0..n-1]$ where n a composite modulus. In this developed scheme the original message and cipher message from the *general linear group* of $h \times h$ matrices over z_n indicated by $g(h, z_n)$ and the original message indicated by m. The message in RSA scheme is encrypted in blocks after divide it to blocks, every block must convert to a value smaller than the modulus n. The intractability of the RSA assumption forms its security. The RSA assumption is the difficulty of solving the integer modulus n, which is a product of two distinct odd large primes p and q with an assistance of another public key e and an integer cipher text c. In other words, the RSA difficulty is that of solving e^{th} roots mod a composite modulus n. The conditions determined the modulus n and the public key e are to guarantee that for every integer $c \in (0,1,...,n-1)$ there is just one $m \in (0,1,...,n-1)$ where $m^e = c \bmod n$.

Advantages of the Improved Scheme

1)The key range of the scheme is considerable. It means that it can be large enough to use by matrices of high level of ranks. The key range for instance in the RSA scheme is of length $q(n) = (p-1)(q-1)$. But, in the

PDF created with pdfFactory Pro trial version www.pdffactory.com

improved scheme the key range is of length $g(n)$.

2)In the Improved scheme, we can employ both a $h \times h$ matrix x and an integer as used in the RSA scheme. The used of a matrix x in fact is not a weakness. It is actually a strength added to the improved scheme, since the RSA scheme is a block cipher. In this case we can adopt h^2 blocks and place them in the matrix x then compute when wanted. So in this case the improved scheme is more flexible compared with the RSA scheme.

Note **:** Assume that $n = p * q$ is the product of two large prime numbers, and suppose that g is the general linear group of $h \times h$ matrices over z_n. Then

$$g = (p^h - 1)(p^h - p)...(p^h - p^{h-1}) * (q^h - 1)(q^h - q)...(q^h - q^{h-1})$$

Cryptographic Hash Function

Uses a mathematical transformation to irreversibly "encrypt" information Hash function, also called message digests and one- way encryption, are algorithms that in some sense, use no key Instead, a fixed-length hash value is computed based upon the plaintext that makes it impossible for either the contents or length of the plaintext to be recovered [15]. Most popular hash function is MD5 which is considered in this work.

MD5 Hash

The MD5 hash also known as *checksum* for a file is a 128-bit value, something like a fingerprint of the file. There is a very small possibility of getting two identical hashes of two different files. This feature can be useful both for comparing the files and their integrity control. Let us imagine a situation that will help to understand how the MD5 hash works. Entity A and Entity B have two similar huge files. How do we know that they are different without sending them to each other? It simply have to calculate the MD5 hashes of these files and compare them

MD5 Hash Properties

The MD5 hash consists of a small amount of binary data, typically no more than 128 bits. All hash values share the following properties:

***Hash length*:** The length of the hash value is determined by the type of the used algorithm, and its length does not depend on the size of the file. The most common hash value lengths are either 128 or 160 bits.

***Non-discoverability*:** Every pair of un-identical files will translate into a completely different hash value, even if the two files differ only by a single bit. Using today's technology, it is not possible to discover a pair of files that translate to the same hash value.

***Repeatability*:** Each time a particular file is hashed using the same algorithm; the exact same hash value will be produced.

***Irreversibility*:** All hashing algorithms are one-way. Given a checksum value, it is infeasible to discover the password. In fact, none of the properties of the original message can be determined given the checksum value alone [16],standard algorithm for MD5 is used in this work.

Authentication System

Authentication is any process, through which one proves and verifies certain information, i.e. determining whether someone or

PDF created with pdfFactory Pro trial version www.pdffactory.com

something is, in fact, who or what it is declared to be. Authentication requires that the information be checked for single, previously identified entity .

Zero Knowledge of Authentication

A zero-knowledge proof or zero-knowledge protocol is an interactive method for one party to prove to another that a (usually mathematical) statement is true, without revealing anything other than the veracity of the statement. A zero-knowledge proof must satisfy three properties [17]:

Completeness: if the statement is true, the honest verifier (that is, one following the protocol properly) will be convinced of this fact by an honest prover.

Soundness: if the statement is false, no cheating prover can convince the honest verifier that it is true, except with some small probability.

International Standard Book Number (ISBN) [18]

The International Standard Computer Number (ISBN) is an identifying number assigned to virtually every software copy. A new edition receives its own ISBN. It serves to uniquely identify the copy. An ISBN for example has four parts:1)Language country code, 2)Manufacturer code, 3)Copy number assigned by publisher, 4)Check digit. For example, a total of 10 digits ISBN 0-387-95045-1 has language/county code 0, publisher code 387, copy number 95045 and a check digit 1. Below lists some language country code.

0 English (UK, USA, NZ, Australia, Canada), English (South Africa, Zimbabwe)

1 French (France, Belgium, Canada, Switzerland)

2 German (Germany, Austria, Switzerland)

3 Japan

4 USSR

5 China

6 India

7 Arabic (All Countries)

It's clear that widely used languages are assigned a short language country codes, thereby allowing for a long publisher code, while other countries and languages have been assigned long language country codes, so their publisher codes must be short. When the number space of language country code is exhausted, another code is assigned to the language country. For example you can give to Spain codes 81 and 93.

Check Digit Calculation

The check digit is computed by multiplying the leftmost ISBN digit by 10, the next digit by 9, and so on up to the ninth digit from the left, which is multiplied by 2. The products are then added, and the check digit is determined as the smallest integer that when added to this weighted sum will make it a multiple of 11. The check digit is therefore in the interval [0, 10]. If it happens to be 10, it is replaced by the Roman numeral X in order to make it a single symbol. If we denote the nine leftmost ISBN digits by d_1 through d_9 (from left to right), then the ISBN I is computed by first calculating the weighted sum:

$$T = (10d_1 + 9d_2 + 8d_3 + 7d_4 + 6d_5 + 5d_6 + 4d_7 + 3d_8 + 2d_9) \bmod 11$$

Notice that T is in the interval [0, 10] because of the use of the mod

PDF created with pdfFactory Pro trial version www.pdffactory.com

and then subtracting $I = 11 - T$. For example, given the nine digit 038795045, the two steps produce.

$$\begin{bmatrix} T = (10*0 + 9*3 + 8*8 + 7*7 + 6*9 + 5*5 \\ + 4*0 + 3*4 + 2*5) \bmod 11 = 241 \bmod 11 = 10 \end{bmatrix}$$

and $I = 11 - 10 = 1$ yielding ISBN 0-387-95045-1

ISBN are assigned, printed, scanned and handled by both machine and humans, so errors can creep in. It is important to detect errors, but there is no need to automatically correct them by means of a sophisticated error correcting code. When the check digit indicates an error in the ISBN, a human can easily identify the error and correct it manually. Obviously, a single check digit cannot detect every error, but it is easy to show that the ISBN check digits can detect all the most common errors. The most common errors in an ISBN are a corrupted digit and two consecutive digits being transposed. It's easy to show that all these errors will be detected by the ISBN check digit [18].

The Details of Proposed software protection system

The general view of the proposed system is shown in figure(1). It begins at the customer's computer where the protected software analyzes the hardware of the client. Then it produces a code which combines several read Hardware serials like CPU and BIOS. The code is encrypted using TDES algorithm. The resulting code will be named "installationID". In the next phase the code will be sent to the development company by either one of two methods:

1. Automatically via internet connection. Either by a registration page or directly without informing the user using a secure model.

2. Manually by phone.

Once the code is received at the development company, the code will be analyzed and then decrypted in order to re-obtain the hardware info of the customer. Now the obtained values will be checked using another provided systems to detect whether the user is installing the system on his machine or on another machine. Now two scenarios has been considered:

1. ***Valid***: The user is installing the software on his machine. Now the company will send a code named "activation code" to the customer. When the user uses this code, the program will run on his machine.

2. ***Not Valid***: If the user is using the software on a different machine then the values will not match and the development company will know that the user is using a pirated copy. No activation code will be sent in this case.

Phase 1: Generate Installation ID

Step 1: Hardware Analyze: At this step an analysis process to Hardware client machine will be get it which are

X: HDD-serial number (15 chars-long).

Y: CPU-serial number (15 chars – long).

Z: BIOS Serial Number: (15 chars-long)

All these component will be used to generate an Machine ID.If there is no internet connection, the client can update the company with the hardware information by telephone or fax, as he will inform them the

PDF created with pdfFactory Pro trial version www.pdffactory.com

installation ID and the Copy ID, which the client will have through the program execution on client's computer.

Step 2: Generate Machine ID

The program will read hardware serial numbers for the CPU, BIOS, and Hard Disk, then generates a unique Machine ID function to CPU, BIOS, and Hard Disk.Machine ID generated by take first 6 character from(X, Y, Z).

Step 3: **encrypt the obtained Machine ID** using TDES Algorithm. The simplest variant of TDES operates as follows: DES(k_3;DES(k_2;DES(k_1;*M*))), where *M* is the message block (here will be the Machine ID) to be encrypted and k_1, k_2, and k_3 are DES keys. See Figure 2.

Phase 2 : Authentication

Activation process takes place by generating and sending an Activation code in case the client has: *Validate Copy ID* (not already used and an error message will appear if the system dictates that), *Validate Installation ID* (Once the company obtains Machine ID, it will check with the company Data Base to detect whether the client is installing the software package on client's machine or on another machine), *Validate IP Address* (our application will consider this validation if and only if the client activates IP Address authentication), and *Validate File Checksum* value (any tampering in the file will lead to failure in the authentication process). At any fail in steps above, company will know that the user is using a pirated copy. No activation code will be sent in this case. Each client could have unlimited number of copies

Generation

The copy ID is mandatory for the activation process as will be seen. Such code can for each purchased application (ex. 10 copies from Microsoft Office). To identify that, a code is added called "Copy ID." This code is unique among items. So each software package could have its own and only copy ID.

Proposed method for copyID

generate using many ways; here Identification Copy Number (ICN) algorithm is proposed method generated through a special algorithm, which an identification number is assigned to virtually every produced copy. It serves to uniquely identify the copy. this proposed algorithm is similar to the one which is used to generate an International standard Number for book(ISBN) described previously. This algorithm is described in Algorithm (1).

Algorithm (1) ICN

Input: *number of N digit which contained(language country code, Manufacturer code, copy number assigned by publisher).*

Output: *identification number for software copy*

1: Start

2: Obtain the number of digits N.

3: Read the digits $d_1, d_2, ..., d_N$

4: Loop on i where i = 1 to (N-1)

a. Compute $T = T + d_i * i$

5: Compute T=T mod 11

6: Compute the digit= 11-T

7: End.

Example :

Language country code 0, publisher code 387, copy number 95045

Digits = 038795045

PDF created with pdfFactory Pro trial version www.pdffactory.com

$$T = (10*0+9*3+8*8+7*7+6*9+5*5+4*0+3*4+2*5) \bmod 11 = 241 \bmod 11 = 10$$

$$I = 11-10 = 1$$

The *check digit = 1*

The identification number for software copy *0-387-95045-1*

§ ***IP Address***

The proposed method will keep using IP Address as an option and up to the clients who are good candidates for IP authentication such as schools, libraries and other organizations that don't go through a common or dynamic IP Address and thus they will have more secure access to activate and run their owned copy and make it much harder to piracy.

§ ***Checksum***

The 128-bit (16-byte) MD5 hashes are typically represented as a sequence of 32 hexadecimal digits. It is extremely unlikely that any two non-identical files existing in the real world will have the same MD5 hash. Even a small change in the message will result in a completely different hash, due to the avalanche effect (When a single bit is changed the hash sum becomes totally different). The checksum calculation and storage will serve as a mean to detect file tampering.

Phase 3: Generate Activation Code

For each client who has a unique Machine ID there is a unique Activation Code in the company Database with the end of transactions between the two parties and will be sent back to the client *only and only if* the Authentication processing is valid. To generate a unique Activation Code, we need to do the following steps:

1) Read hardware serial numbers for the CPU, BIOS, and Hard Disk.
2) Generate Machine ID, then,
3) Encrypt step 2 using Triple DES, then,
4) Hash step 3 using MD5 hash.

Protection Technique (EXE Encapsulator)

EXE Encapsulator is a program while will encapsulate any existing "EXE" file without having an access to its internal source. The encapsulated file will use methods to validate the authenticity of the copy and protect it from any non-authorized execution. This program is very helpful if the method is used with currently developed programs, which do not have its source code. The Encapsulator is proposed method for software protection, which described in Algorithm(2) which is create a file called it (protect.exe) by combining three files into one protected file and MD5 calculation to generate the checksum file through the encapsulation process and kept is hidden as this file will be part from the protected exe file. The Encapsulator combines the given executable with the protection interface which is stored externally on a file named "protect.exe". The protection interface and its required libraries shall be presented at the same folder as the required executable. When combining the files, the resulting file will run to show protection interface which is responsible for the logic behind the authentication process. The *Protected File Structure* is a combination of three files :

PDF created with pdfFactory Pro trial version www.pdffactory.com

1) **Protect.exe**: a file which contains the protection interface (protection technique).

2) **File.exe**: Exe file is used to install and run the program and routines. If it is wanted to protect this file (applying the protection techniques to), it might take any name ending with the extension ".exe."

3) **Checksumsfile.txt**: a file which contains the checksum values of "file.exe". This file is generated during the encapsulation process.

The resulting file will have the same name"file.exe" and will replace the existing file.exe

Algorithm (2) Encapsulator

Input *:(the file.exe without protection technique)*

Output*: the protected file*

1: Start

*2: Read the contents of the first file (**Protect.exe**),then, Write the contents into the output file (**Protect.exe**).*

*3: Read the contents of the second file **(file.exe)**.,then, Write the contens into the output file **(file.exe)**.*

*4: Read the contents of the third file (**Checksums file.txt**,then, Write the contents into the output file (**Checksums file.txt)**.*

*5: Joining files in directory into one file (**Output protected file**).*

6: Get checksum by invoke MD5 function.

7:write the checksum file to the text file

8:End.

Protection interface through two paths which the protection interface works are:

Path 1 (Manual Mode)

In this path, the program will go through the following operations, but we will consider that; there is no internet connection, and, no authentication process (which means there is no actions for the Zero Knowledge protocol).The operations are:*Generate installation* ID, and activation code using TDES and MD5 hash, *The activation code* comparison with the company database, *Read the stored* checksum then calculate File Checksum, and do a checksum comparison .

Path 2 (Automatic mode)

In this path, the program will go through the following major operations, but here we will consider there is an internet connection, The file reg.ini in its place, a correct value for the user name and password, and a successful authentication process.

Generate installation ID, and activation code using TDES and MD5 hash.

The activation code comparison .

Read the stored checksum then calculate File Checksum and do a checksum comparison

Authenticate the user using the Zero Knowledge protocol.

Exchange data using the improved RSA algorithm, using improved RSA algorithm which is used to encrypt machine ID and ensure the security of system against a replay attack by achieving the random communication(encrypted machine ID) to the application so that cracker cannot discern legitimate from illicit communication between the client's machine and the development machine **Customer Tracking System**

This is a small system, which stores information about the users and their equivalent hardware serials. The

PDF created with pdfFactory Pro trial version www.pdffactory.com

system will be used to identify whether the user is legal or not.

1. Website

Register New User page: Create a typical set of information record for each client, **Login page:** Access to the Database using the Protocol Authentication the Zero-Knowledge Authentication, **Products page:** List all the client's activated products, and, **New Copy activation page:** a sub page from the Products page.

2. Webserivce: The essential part of Web services is the *Interact* relationship between a Service provider and Service requestor. To get some programming task done, that can make use of a web service by calling it over the Internet. By passing parameter data with the request, it can be expected to receive a response containing the result generated by the web service. Web Sites are just the user interface of this application… and Web Service is intended to expose some functionality to the outside or to some other layer as a service. The web service is used in the path where the protection interface needs to interact with the server directly using the secure protocol (Zero-Knowledge Authentication) in order to activate the copy; in which the client is directly connected to the net and has a reg.ini file in his directory in addition this is the case in which the client has already sent his pc to the company.

3. Database: stores the information about the clients and their purchased products. This database can be accessed through either the web service or through the web interface.

Implementation and Practical Example :

The proposed system for software protection is implemented on Pentium 4 PC computer with CPU of 1.6 GB and RAM of 1GB on Window XP using VB.net and C# programming language. And the following examples are to teste practically the proposed in order to prove its functionalities.

Example1 (*Manual mode/we)b*:

In this example the proposed system executed at *Manual mode/web* tests. At table 1 the system condition in the case Invalid Activation Code, Table2 shows case 2 where The original protected file executed successfully, Table 3 shows system Conditions for case 3 file has been tampered ,Table 4 System Conditions for case4 Web error: Invalid Copy ID. Table 5 System Conditions for case 5 Web error: This copy is registered for different user. Table 6 System Conditions for case 6 The application executes.

Example2 (*Automatic Mode /Web Service Test) :*

In this example the proposed system executed at *Automatic Mode /Web Service Test.* Table 7 System Conditions for case 1 in the automatic mode, Expected result No User information files were stored. The system shall fail with the automatic operation and proceed to the manual mode. Table 8 System Conditions for case 2 The protected program worked normally, expected Result The system will fail to authenticate the user and will proceed to the manual mode. . Table 9 System Conditions for case 3 the protected program worked normally. ,Table 10 System Conditions for case 4 using a different IP ,Table 11 System Conditions for case 5 change

PDF created with pdfFactory Pro trial version www.pdffactory.com

the original user's hardware change the original user's hardware the Expected Result The user did change his original hardware. Thus the system will not authorize the software execution. Table 12 System Conditions for case 6 System work with different

The Evaluation of the Proposed System

The every part of the proposed system has been solved, the most common types of software piracy problem are illustrated in following points:

The End-user piracy (single licensed copy is used and installed on multiple computers): is solved by validating *Installation ID* that is checked with the system database to detect whether the software for specific user is on specific machine (using TDES algorithm and MD5 algorithm).see Table (5-6).

The End-user piracy (software is duplicated then distributed):is solved by validating *Copy ID* by using ICN algorithm .see Table (5).

Client-server piracy:(represents customer's machine and development's company server) is solved in the online mode of the proposed system by using reg.ini file by applying Zero knowledge Authentication to validate use name and password on registration page for the protection system. In addition, Improved RSA algorithm is used to ensure that the exchange data is not attacked by encrypting the data. See Table (9).

Internet piracy: is solved by validating *copy ID* (not already used) and *validating IP* address (if the client activates IP Address Authentication).see Table (11).

Hard disk loading piracy : is solved by validating *installation ID(*which is ensures that specific copy is to specific machine) and validating *copy ID* which checks the copy ID in the database of the system which represents the company development database using ICN algorithm .see Table (12).

Counterfeiting (duplicating and selling copyrighted software):is solved by validating *Installation ID* and validating *copy ID(*the specific copy for specific machine).

The proposed system has the ability to detect any tampering on the protected exe file as the cracker tries all the time to do some modification on the main exe file for any application and tries to execute it through that.

Any modification on the protected exe file in the proposed system means that the size of the exe file will change and thus the MD5 hash value will change accordingly and even the change is so simple the MD5 hash value(checksum value) will be completely different (see Table (3).

The proposed system is secure against known threats attack (Replay attack, brute force attack and man-in-the middle attack). *Replay attack:* The proposed system is protected against this type of attack that sends random communication (Encrypted Machine ID) using the improved RSA algorithm.

Brute force Attack (password-guessing attack): The proposed system is protected against this type of attack by authenticate the user using Zero Knowledge Authentication (without knowing the

PDF created with pdfFactory Pro trial version www.pdffactory.com

user's password) that does not store the password in the system.

Man- in –The- middle attack (by Hardware Replacement): The user did change the original hardware .Thus the proposed system will not authorize software execution.

Conclusions

The following conclusions can be derived from this work:

1. The proposed method Software Protection has been solved the most common types of software piracy problem showed in tables (5, 4, 8, 10, 11).
2. The proposed method Software Protection is secure against known threats attack (Replay attack, Brute force attack, and Man-in-the Middle attack) by using the improved RSA and MD5 and the mixture of cryptography techniques respectively.
3. The implemented system is efficient. It provides no execution time penalty on the protected programs, see Table2.
4. The implemented system is dynamic. It can be easily applied to any (.net) based software without the need for its source code.
5. It is a scalable software based method that does not bind the user to a specific version of the operating system,see Table 12.
6. The implemented system serves all kinds of users. Home users who have no network connection, home users who have no permanent internet connection, high security enterprise users who should execute their applications only from a specific location, and other users who have mixed conditions. That is shown in Manual mode and Automatic mode.
7. A proposed method to protect all the system files. The proposed software protection method described in Algorithm (2).

References

[1] Cowan, C., "**Software Security for Open-Source Systems, IEEE Security and Privacy**", Vol(01),pp.38-45,2003.

[2] T.Premkumar Devanbu and Stuart Stubblebine, "**Software engineering for Software Engineering for security a roadmap**" ,In Proceeding of the conference on the future of software Engineering,pages 227-239.ACM Press,2000.

[3] IDC, "**BSA/IDC Global Software Piracy Study**", Transitions Online, 2007.

[4]BSA website **http://www.bsa.org/country/Anti-Piracy/What-is-Software-Piracy/Types%20of%20Piracy.aspx**, 2007.

[5] StarForce. **http://www.star-force.com/**, 2007.

[6] B.Anckaet, Bjorn De Sutter and Koen De Bosschere, "**Software Piracy Prevention through Diversity**",Proceedings of the 4th ACM workshop on digital rights management, p.63-71,2004.

[7] Bruce Schneier. " **Schneier on Security: Sony's DRM Rootkit: The Real Story**", http://www.schneier.com/blog/archives/2005/11/sonys_drm_rootk.html, 2005.

[8] "**Concepts of Cryptography**," URL:http://www.Kremlinencrypt.com/concepts.htm.

[9] "**Encryption Algorithms**",URL :

PDF created with pdfFactory Pro trial version www.pdffactory.com

http://www.networksorcery.com/enp /data/encryption.htm.

[10] Wikipedia, "**The Free Encyclopedia, Data Encryption Standard**," URL: http://www.wikipedia . 2(28), 44-60, 1997.

[11] Smith,Don Copper,"**The Data Encryption Standard (DES) and its**

Strength Against Attacks", IBM Journal of Research and Development

(1994), 38(3), p243-250.

[13] Zoeller & Renate. " **Nest of Pirates**", Transitions Online, p.5-5, 2007.

[14] Mustafa Al-Fayoumi, Sattar J. "**An Efficient RSA Public Key Encryption Scheme**"**,** Information Technology: New Generations, 2008. ITNG 2008. **Fifth International Conference**, p.127-130, 2008.

[15] Stallings, "**Cryptography and Network Security :Principles and Practice**". , Second Edition. Upper Saddle River, NJ: Prentice Hall,1999

[16]Wikipedia,**http://en.wikipedia.org/wiki/Md5**, 2007.

[17] Bruce Schneier. **Applied Cryptography**, 2nd ed., p.312, p.415, John Wiley & Sons, 1996.

[18] David Salomon, "**Coding For Data And Computer Communications**", Springer, 2005.**International Conference**, 2(4), 816-819, 2004.

PDF created with pdfFactory Pro trial version www.pdffactory.com

Table (1) System Conditions in the case1 Invalid Activation Code

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**False**
IP	**-**
Stored IP	**-**
Current IP	**-**
Copy ID	**0-387-95045-1**
Copy ID is valid and in data base	**Yes**
Copy ID is used for different user	**No**
Copy ID is used for the same user	**No**
Installation ID	**ASQl46NXVpU=-ZZN+JbqHj38=-cT80sfpbkes=**
Required activation Code	**CNa/Y82FqI27+lz1ZC3b5w==**
Providd activation code	**A**
Reg.ini file available	**No**
Tampered on purpose	**No**
Internet connection	**No**

Table (2) System Conditions for case 2 The original protected file executed successfully

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**False**
IP	**-**
Stored IP	**-**
Current IP	**-**
Copy ID	**0-387-95045-1**
Copy ID is valid and in data base	**Yes**
Copy ID is used for different user	**No**

PDF created with pdfFactory Pro trial version www.pdffactory.com

Copy ID is used for the same user	No
Installation ID	ASQl46NXVpU=-ZZN+JbqHj38=-cT80sfpbkes=
Required activation Code	CNa/Y82FqI27+lz1ZC3b5w==
Provided activation code	CNa/Y82FqI27+lz1ZC3b5w==
Reg.ini file available	No
Tampered on purpose	No
Internet connection	No

Table (3) System Conditions for case 3 file has been tampered

Hard Disk Serial Number	3753566138
BIOS Serial Number	CNF5141SNX
CPU Serial Number	AFE9FBFF000006D8
IP condition	False
IP	-
Stored IP	-
Current IP	-
Copy ID	0-387-95045-1
Copy ID is valid and in data base	Yes
Copy ID is used for different user	No
Copy ID is used for the same user	No
Installation ID	ASQl46NXVpU=-ZZN+JbqHj38=-cT80sfpbkes=
Required activation Code	Can/Y82FqI27+lz1ZC3b5w==
Provided activation code	Can/Y82FqI27+lz1ZC3b5w==
Reg.ini file available	No
Tampered on purpose	Yes
Internet connection	No

PDF created with pdfFactory Pro trial version www.pdffactory.com

Table (4) System Conditions for case Web error: Invalid Copy ID

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**False**
IP	**-**
Stored IP	**-**
Current IP	**-**
Copy ID	**0-487-95045-1**
Copy ID is valid and in data base	**No**
Copy ID is used for different user	**No**
Copy ID is used for the same user	**No**
Installation ID	**ASQl46NXVpU=-ZZN+JbqHj38=-T80sfpbkes=**
Required activation Code	**CNa/Y82FqI27+lz1ZC3b5w==**
Provided activation code	**CNa/Y82FqI27+lz1ZC3b5w==**
Reg.ini file available	**No**
Tampered on purpose	**No**
Internet connection	**No**

Table (5) System Conditions for case 5 Web error: This copy is registered for different user.

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**False**
IP	**-**
Stored IP	**-**
Current IP	**-**
Copy ID	**0-387-95045-1**
Copy ID is valid and in data base	**Yes**

PDF created with pdfFactory Pro trial version www.pdffactory.com

Copy ID is used for different user	Yes
Copy ID is used for the same user	No
Installation ID	ASQl46NXVpU=-ZZN+JbqHj38=-T80sfpbkes=
Required activation Code	CNa/Y82FqI27+lz1ZC3b5w==
Provided activation code	CNa/Y82FqI27+lz1ZC3b5w==
Reg.ini file available	No
Tampered on purpose	No
Internet connection	No

Table (6) System Conditions for case 6 The application executes.

Hard Disk Serial Number	3753566138
BIOS Serial Number	CNF5141SNX
CPU Serial Number	AFE9FBFF000006D8
IP condition	False
IP	-
Stored IP	-
Current IP	-
Copy ID	0-387-95045-1
Copy ID is valid and in data base	Yes
Copy ID is used for different user	No
Copy ID is used for the same user	Yes
Installation ID	ASQl46NXVpU=-ZZN+JbqHj38=-cT80sfpbkes=
Required activation Code	CNa/Y82FqI27+lz1ZC3b5w==
Provided activation code	CNa/Y82FqI27+lz1ZC3b5w==
Reg.ini file available	No
Tampered on purpose	No
Internet connection	No

PDF created with pdfFactory Pro trial version www.pdffactory.com

Table (7) System Conditions for case 1 The manual activation dialog were displayed.

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**True**
Stored IP	**192.168.1.3**
Current IP	**192.168.1.3**
Reg.ini file available	**No**
Tampered on purpose	**No**
Internet connection	**Yes**

Table (8) System Conditions for case 2 The manual activation dialog was displayed.

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**True**
Stored IP	**192.168.1.3**
Current IP	**192.168.1.3**
Reg.ini file available	**Yes with invalid user information**
Tampered on purpose	**No**
Internet connection	**Yes**

PDF created with pdfFactory Pro trial version www.pdffactory.com

Table (9) System Conditions for case 3 The protected program worked normally.

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**True**
Stored IP	**192.168.1.3**
Current IP	**192.168.1.3**
Reg.ini file available	**Yes**
Tampered on purpose	No
Internet connection	Yes

Table (10) System Conditions for case 4 using a different IP

Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**True**
Stored IP	**192.168.1.3**
Current IP	**204.168.1.4**
Reg.ini file available	**Yes**
Tampered on purpose	**No**
Internet connection	**Yes**

PDF created with pdfFactory Pro trial version www.pdffactory.com

Table (11) System Conditions for case 5 change the original user's hardware change the original user's hardware

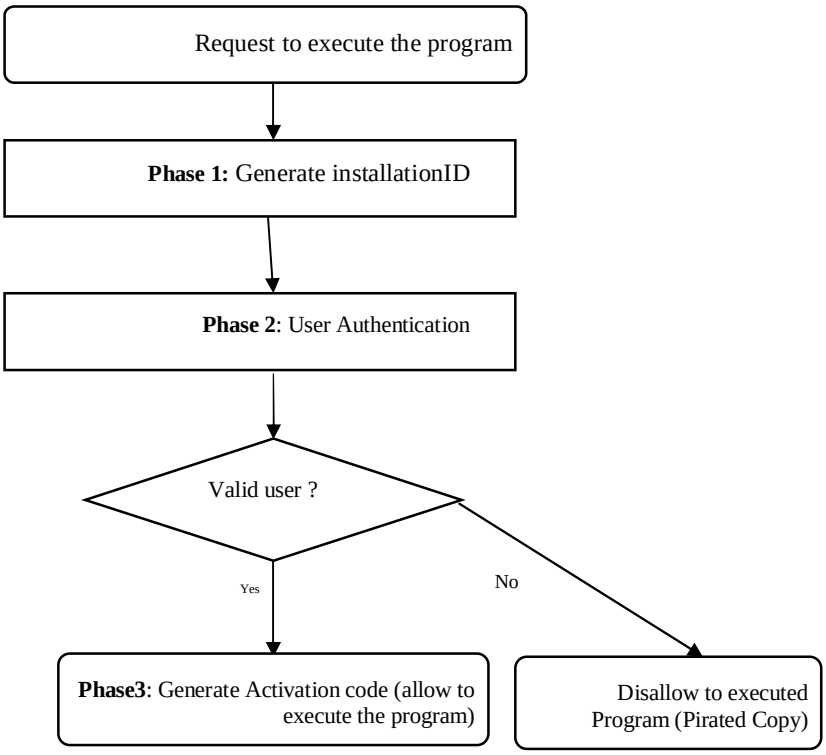
Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CNF5141SNX**
CPU Serial Number	**AFE9FBFF000006D8**
IP condition	**True**
Stored IP	**192.168.1.3**
Current IP	**192.168.1.3**
Hard Disk Serial Number	**3753566138**
BIOS Serial Number	**CCC5141SNX**
CPU Serial Number	**AFE9FBDDD06D8**
Reg.ini file available	**Yes**
Tampered on purpose	**No**
Internet connection	**Yes**

PDF created with pdfFactory Pro trial version www.pdffactory.com

Table (12) System Conditions for case 6 System work with different version of Windows Xp

Hard Disk Serial Number	**4131583554**
BIOS Serial Number	**6FKDJ2J**
CPU Serial Number	**BFE9FBFF000006E8**
IP condition	**False**
IP	**-**
Stored IP	**-**
Current IP	**-**
Copy ID	**0-387-95045-1**
Copy ID is valid and in data base	**Yes**
Copy ID is used for different user	**No**
Copy ID is used for the same user	**No**
Installation ID	**ASQl46NXVpU=-ZZN+JbqHj38=-cT80sfpbkes=**
Required activation Code	**CNa/Y82FqI27+lz1ZC3b5w==**
Provided activation code	**CNa/Y82FqI27+lz1ZC3b5w==**
Reg.ini file available	**No**
Tampered on purpose	**No**
Internet connection	**No**

PDF created with pdfFactory Pro trial version www.pdffactory.com



Figure(1) proposed system general view

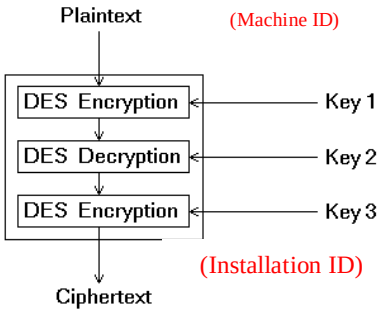


Figure (2) Installation ID Generation using TDES

PDF created with pdfFactory Pro trial version www.pdffactory.com