Copyright Protection of Online Application using Watermarking

B. K. Sharma
Asstt. Prof., Deptt. Of
Computer Application,
AKGEC, Ghaziabad, India

R. P. Agarwal
Professor and Dean Research
& Academics, Shobhit
University, Meerut, India

Raghuraj Singh
Professor, Department of
Computer Science & Engg.,
HBTI, Kanpur, India,

ABSTRACT

Today Intellectual Property Rights (IPR) for online application and software code has become a fundamental issue. A variety of prevention techniques have been developed for copyright protection of software codes or intellectual property rights or web application. But, unfortunately no single technique is currently strong enough to protect the software codes. However, through a combination of techniques software developer can better protect their software codes. The combination of watermark techniques is visible watermarking and invisible watermarking. The visible watermarking is static watermarking techniques where as the invisible watermarking is a dynamic watermarking technique. In the static watermarking techniques, the watermark is stored in the data section and code section both where as in the dynamic watermarking techniques the watermarks are stored during program execution through native codes. In this paper, we have proposed dual watermarking techniques for an online application using Java Virtual Machine. The dual watermarking technique is a combination of both a visible and an invisible watermark.

Keywords

Intellectual Property Rights (IPR), Dual Watermarking Technique (DWT), Visible watermarking, Invisible watermarking

1. INTRODUCTION

Software piracy means illegal use or distribution of software codes. It is one of the main direct threats to software industry, which will bring serious damages to the interests of software developers or providers. As per the survey released by the Business Software Alliance [4] the rates of illegal use of software's of different years of countries like Asia Pacific, Central and Eastern Europe (C & E Europe), Latin America, Middle East and Africa (ME & Africa), North America (N America) and Western Europe (W Europe) are shown in table-1 and its comparison bar chart is shown in Figure 1:

Country Name	Piracy Rate (%)						
	2003	2004	2005	2006	2007	2008	2009
Asia Pacific	53%	53%	54%	55%	59%	61%	59%
C & E Europe	71%	71%	69%	68%	68%	66%	64%
Latin America	63%	66%	685	66%	65%	65%	63%
ME & Africa	56%	58%	57%	60%	60%	59%	59%
N America	23%	22%	22%	22%	21%	21%	21%
W Europe	36%	34%	35%	34%	33%	33%	34%

Table – 1: Piracy Rate in %

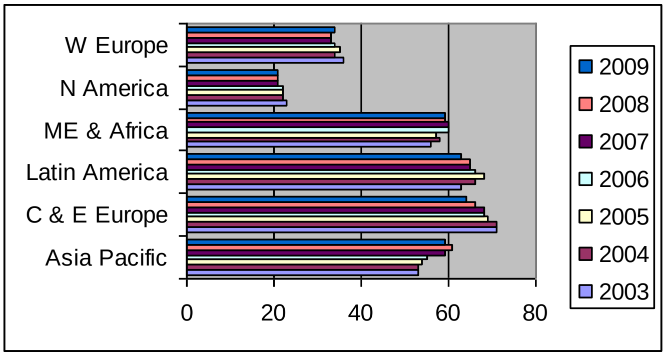


Figure-1: Piracy Rate in %

It means privacy of software directly affects the revenue of software vendors. The software vendors are losing every year million of US dollars. Again as per the survey released by the Business Software Alliance [4] the software vendors have lost every year million of US dollars We have shown some countries data like Asia Pacific, C & E Europe, Latin America, ME & Africa, N America and W Europe from last seven years in Table-2 and its comparison bar chart is shown in Figure-2. It is increasing day by day due to the much easier use of Internet. . It is increasing day by day due to the much easier use of Internet. As a prevention technique, one of the most promising attempts to protect intellectual property rights includes software watermarking. Software watermarking is a new research area that aims at providing copyright protection for commercial software. It is relatively new software protection technique appeared in recent decade, whose basic principle is to embed secret information as the evidence to identify an ownership and track pirated software. This technique is also used in other kinds of protection and enforcement of intellectual property rights such as text, digital images, digital audio, digital video etc. The software can be protected by two main approaches namely hardware-based and software-based [3] techniques. In this paper we focus on only software based watermarking techniques.

Country Name	Amount ($)						
	2003	2004	2005	2006	2007	2008	2009
Asia Pacific	7555	7897	8050	11718	14090	15261	16544
C & E Europe	2111	2682	3262	4124	6351	7003	4641
Latin America	1263	1546	2026	3125	4123	4311	6210
ME & Africa	1018	1239	1602	1985	2446	2999	2887
N America	7245	7549	7686	8104	9144	10401	9379
W Europe	9612	1185	11856	10642	11655	13023	11750

Table – 2 : Losses amount in $

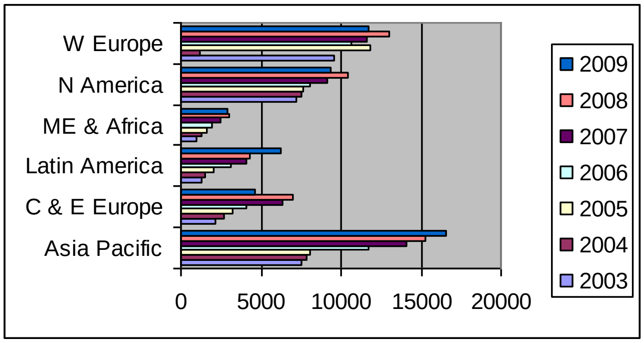


Figure-2: Losses amount in $

In the software-based technique, the developers or providers used earlier registration codes, license files, shelling of the codes and many other methods, which protect the software merely via the software itself. The most common implementation technique is to put the registration on the client. It requires a legal token to give the user permission to use. The token may be a license key, a license file or an activation code and so on. But for online application java is the most popular programming languages for software development. Java is a high portable due to the Java byte code technique and Java Virtual Machine. It is basically compile once and run everywhere. The byte code provide detailed information about code and make it easy to decompose into reusable class files and decompiled into source code by malicious users.

The software codes are copied by most of the people due to the following reasons [7].

- Software is intangible and/or non-exclusive
- Everyone does it
- It is very easy to copy software codes
- It does not harm anyone
- The low quality of software
- Software is expensive
- The risk is minimal

2. TECHNIQUES OF COPYRIGHT PROTECTION

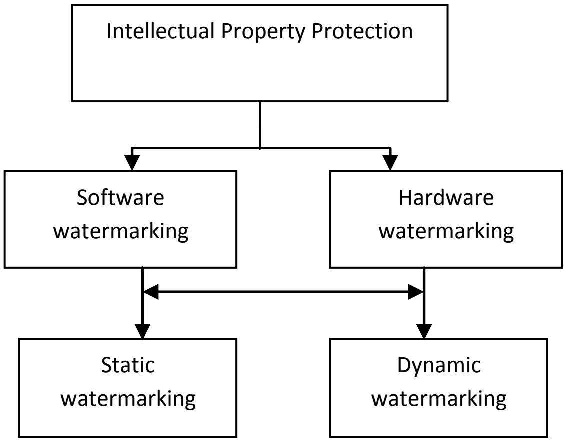
Various techniques of copyright protection of software codes have been defined. These techniques are categorized into two ways: static watermark and dynamic watermark.

2.1 Static and Dynamic watermarking

In static watermarking the watermark [10] is stored in the source code, either in the data section or in the code section. The one kind of static watermarking for application software is naming convention like variable name always starts with VAR or numeric is appended at the end of the variable name [2]. Infosys Company is using these concepts [9]. This company is using these concepts at the time of declaring variables. In each variable first letter always starts with their data types like integer variable start with letter i, float variable starts with letter f, double variable start with letter d etc [6]. The other static watermarking recursively applies mathematical operations on a variable, which has no effect in over execution of the program. The extraction of such watermarks does not need to run the software.

The other static watermarking techniques used to extract code from the software code not only include some critical functional codes, but also some critical resources such as text information [5].

In dynamic watermarking, the watermarks are generated during program execution and stores in the program execution states. The dynamic watermarking hides the watermark in data structures that is built specially for embedding purpose during execution of the program.



The Semblance Based Disseminated Software Watermarking Algorithm (SDSW) [2] describes the watermarking techniques for the virtual environment like JVM. This is based on java-based applications. SDSW is designed to encode the secret information, which is added to the program after compilation. This encoding of secret information within the program is achieved by adding dummy instructions, which are hard to identify and to replace. The SDSW is divided into four steps [2]. (a) Watermark Encoding (b) Dictionary Mapping (c) Instruction Embedding (d) Watermark Recognizer.

The other dynamic watermarking techniques are based on hash functions in which the watermark is embedded. Let the watermark to be embedded W is mapped into a large integer at first and divided into many small parts w1, w2, …, wn based on these parts. A hash function is constructed in the following form:

H(x) = wi where i = 1, 2, …, n

The hash function H(x) is embedded into the program being protected.

3. WATERMARK ENCODING TECHNIQUE FOR ONLINE WEB APPLICATION

3.1 Static Watermark Encoding Technique

Let us assume that “Not allowed symbol please use valid symbol” is a static watermark and we want to embed it into online web application then we can embed it with different exceptions that represent authentication and validation results in a web page for online web application. The different exceptions are defined as follows:

```
String waterm1 = "Not allowed symbol @ in user
name please use valid symbol ";
String waterm2 = "Not allowed symbol # in user name
please use valid symbol ";
String waterm3 = "Not allowed symbol $ in user name
please use valid symbol";
String waterm0 = "Not allowed symbol please use
valid symbol";
String waterm4 = "Not allowed symbol % in user
name please use valid symbol";
String waterm5 = "Not allowed symbol ^ in user name
please use valid symbol ";
String waterm6 = "Not allowed symbol & in user
name please use valid symbol ";
String waterm7 = "Not allowed symbol < in user name
please use valid symbol ";
String waterm8 = "Not allowed symbol > in user name
please use valid symbol ";
```

To implement the above watermark we use JAVA code and JAVA is a purely object oriented language i.e. based on class and object. Through this feature we can encapsulates data and methods and by the inheritance we argued that the complexity of a class grow with its distance from the root and the number of its direct descendants. The three steps required to embed the watermark for web based online application.

1. Classes will split into a set of abstract class and a concrete class. These set of abstract class and concrete class are organized in such a manner so that they represent a hierarchical tree structure so that the concrete class extends abstract classes and the top-level abstract class implements the interface.
2. We write a combination of methods that contain the watermark or the exception messages shown above. For example, the methods could be as follows:

```
public String watermark1(int d) {
String waterm1 = "Not allowed symbol ! in
user name please use valid symbol ";
return waterm1;
}
public String watermark2(int d){
String waterm2 = "Not allowed symbol @ in
user name please use valid symbol ";
return waterm2;
}
.....
public String watermark0(int d) {
String waterm0 = "Not allowed  symbol
please use valid symbol ";
return waterm0;
}
.....
```

3. We extend these methods to the abstract and concrete classes generated in step first. Through all these three steps, the watermark will hide among a number of exceptions and spread in a hierarchical tree structure, which is very hard to be distinguish for attacker.

3.2. Dynamic Watermark Encoding Technique

The dynamic watermark is used to secure the key value when it is input from the web page to trace the watermark. In this technique we have combine the Java code and native code written in C language to protect the key when it is input from the web page to trace the watermark. We have proposed first web page as login page where the user input the key through user ID and password. After that when we can select other pages to input the key first of all in login page, the different authentication and validation require. The different authentication and validation will be undertaken through exceptions, so that it hides the key among these results and makes it hard for attackers to distinguish. We have also assume that the user ID does not contain special symbols like @, $, #, ^, & etc. if such character exists in user ID the exception will be thrown.

Generally, the key is unique and easy to distinguish but to make secure we have use C language to confuse the key and other input because C language code will be compiled to machine language directly and cannot be decompiled to source code. It can only be analyzed from assemble language. The method written in C language is called native method. J2SE provides the Java Native Interface (JNI) to access the native methods [8].

3.3. Dual Watermark Encoding Technique

The dual watermark technique is a combination of both visible and an invisible watermark [1]. The visible watermarking is static watermarking techniques in which the watermark is stored in the data section and code section both where as the invisible watermarking technique is a dynamic watermarking technique in which the watermarks stored during program execution through native codes. In this paper we have combined the both visible or static and invisible or dynamic watermarks together to improve the robustness of online web application which is explained in section 3.1 and 3.2.

4. DICTIONARY MAPPING FOR ONLINE WEB APPLICATION

Dictionary mapping maps the set of possible dummy instructions or variables, which are inserted in the program to encode watermark. In this technique we will maps the all the special symbols with random generated value by native code at time of giving user ID and password by one to one function.

5. INSTRUCTION EMBEDDING FOR ONLINE WEB APPLICATION

Instruction Embedding explains various techniques for watermarking like watermark the whole class, every method or selected methods. In this technique we have generated a random number for each special symbol through C language and by hash function again we have generated new number for the same special symbol. Each special symbol generates a integer value from a given random and hash function. Now we can embed the watermark among these integers value and return an integer to the java class. Table-3 shows the encoding format. We have taken only an integer value because only an integer value may represent authentication and validation result. This is a return from the native code program to the java class, it is impossible for attackers to distinguish which integer represents the key. The following pseudo-code implemented by native code return as a result, output value, a random integer, function, is returned to java class.

```
integer result;
if useID contains "@"
        randomvalue1 = random(0,110);
        randomvalue2 = random(111,899);
        result = randomvalue1 + 2 * randomvalue2
– (randomvalue1 +  randomvalue2);
else if userID contains "#"
        randomvalue = random(900,1101);
        randomvalue = random(1102,1732);
        result = randomvalue1 + 2 * randomvalue2
– (randomvalue1 +  randomvalue2);
else if userID contains "$"
        randomvalue1 = random(1733,2222);
        randomvalue2 = random(2223,4723);
        result = randomvalue1 + 2 * randomvalue2
        – (randomvalue1 +  randomvalue2);
else if userID contains "%"
        randomvalue = random(4724,4900);
        randomvalue = random(4901,4999);
        result = randomvalue1 + 2 * randomvalue2
        – (randomvalue1 +  randomvalue2);
……………………………
…………………………….
```

Symbol	@	#	$	%	^
Integer (x)	111 - 899	1102 - 1732	2223 - 4723	4901 - 4999	5112 - 5300
Function	F(x)	F(x)	F(x)	F(x)	F(x)

Table-3 (i)

Symbol	&	<	>	!
Integer (x)	5621 - 5755	5912 - 6125	7145 - 7655	7895 - 7999
Function	F(x)	F(x)	F(x)	F(x)

Table-3 (ii)

6. WATERMARK RECOGNIZER FOR ONLINE WEB APPLICATION

Watermark Recognizer recognizes the method to extract the watermark by tracing the dummy instructions. The dummy instructions are scrutinized by scanning the program and each time instruction counterparts to the one in dictionary its corresponding binary is recorded. In this method we have to input the pre-defined key in the login page to retrieve the embedded watermark. Since the key is an integer, which lies between the range. After the user input a username or the pre-defined key in the login page, Java program will validate the input and then return an integer according to different exceptions that includes the case of inputting a illegal username or the key. This is achieved by a piece of C language code that is introduced in Instruction Embedding for Online Web Application. According to the returned integer, different error messages or the embedded watermark will be returned to the user. The following pseudo-code shows how an error message or the watermark is picked up and sent to the client side according to the returned integer.

```
result object.validate(input)
if result is between 111 and 899
outputString = generating_object.watermark1();
else if result is between 1102 and 1732
outputString = generating_ object.watermark2();
else if result is between 2223 and 4723
outputString = generating_ object.watermark3();
else if result is between 4901 and 4999
```

```
        outputString = generating_ object.watermark4();
        ........................
        ………………..
```

The methods like watermark1( ), watermark2( ), and watermark0( ) etc are defined in section 3.1. These methods are spread among a hierarchy of abstract classes and concrete class. To confuse the attacker even we can introduce some more dummy methods and these dummy methods do not perform any things in the hierarchy tree structure. The dummy methods are:

```
        public String watermark9( ) {
                String s="Successful ";
                return s;
                }
        public String watermark10( ) {
                String s="More successful";
                return s;  }
```

7. THREAT ANALYSIS FOR ONLINE WEB APPLICATION

The attacker can attack on online web application in following main three categories [3]:

Subtractive attacks: The subtractive attacks try to remove the watermark from the software codes. In the proposed methods if attackers remove the watermark, then the some functionalities of the program will damage for example if our define function F(x) will remove from code then the result will be different from original.

Distortion attacks: The distortion attacks will not remove the watermark but might be able to damage or distort it in a way that the owner cannot prove the ownership of the software codes. In the proposed methods if attackers will change the range of my defined random function then we can identify the changes by another random function.

Additive attacks: The additive attacks can insert own watermarks in the software codes. The new watermark can either replaces the original watermark or be inserted in addition to the original watermark and thus it would be difficult to prove which watermark was inserted first. In the proposed methods a sign of the program is set up in advance, and at running time, a new sign is generated and compared with the original one. Any difference between these two signs can prove the program is modified

The proposed technique is robust due to its dual watermark behavior and it is very useful to show the ownership or an intellectual property right of an online web applications.

8. CONCLUSION

In this paper, we have proposed a dual watermarking scheme for the copyright protection of online application using watermarking. Through this technique we can improve the functioning of the online applications. In this techniques we have hide the watermark among various exceptions and the methods. The exceptions and the methods are used to return the watermark or exceptions in a hierarchy of abstract and concrete classes. The key for watermark retrieval is also secured by a native method written in C language. The software source codes are one of the most important assets of the information society, and thus it is important to capture, store and apply it without any piracy. These are the major purposes of software source codes. Thus, software source code management refers to the process of capturing or creating source code, storing and protecting it, updating it constantly, and using it whenever necessary.

9. REFERENCES

[1] Ibrahim Kamel,Qutaiba Albluwi, “A Robust Software Watermarking for Copyright Protection” Elsevier, Computers and Security, January 2009, PP 1-24

[2] Zeeshan Pervez, Noor-ul-Qayyum, Yasir Mahmood, Hafiz Farooq Ahmad, “Semblance Based Disseminated Software Watermarking Algorithm” IEEE 23rd International Symposium on CIS, 27-29 Oct. 2008, PP 1-4

[3] Xuesong Zhang, Fengling He, Wanli Zuo, “Hash Function Based Software Watermarking” IEEE International conference on ASEA 2008, 13-15 Dec. 2008, PP 95 - 98

[4] Business Software Alliance. http://www.bsa.org

[5] Yawei Zhang, Lei Jin, Xiaojun Ye, “Software Piracy Prevention: Splitting on Client”, IEEE International Conference on Security Technology, December 13-15, 2008

[6] Siva subramanyam Y, Deepak Ranjan Shenoy, “Computer Hardware and System Software Concepts” Vol -1, Version 1.0, March 2007

[7] Mikko T. Siponen, Tero Vartiainen, “Unauthorized Copying of Software - An Empirical Study of Reasons For and Against”, SIGCAS Computers and Society, Volume 37, No. 1, June 2007, PP 30 -43.

[8] Ronghui Tu, Feiyuan Wang, Jiying Zhao, Abdulmotaleb El Saddik, "Copyright Protection of Web Applications through Watermarking," icicic, vol. 3, pp.78-82, First International Conference on Innovative Computing, Information and Control - Volume III (ICICIC'06), 2006

[9] Petar Djekic, Claudia Loebbecke, “Preventing application software piracy: An empirical investigation of technical copy protections”, The Journal of Strategic Information System, Volume 16, Issue,June 2007, PP 173-186.

[10] B. K. Sharma, R.P. Agarwal & Raghuraj Singh “An IPR of software codes using watermarking for Modular Programming”, IJMCS, January- June -2010, PP 55 - 58