

Software Piracy Protection

Mrs. D. Seema Dev Aksatha¹ MCA, M.Phil, M. Blessing Marshal²

¹Associate Professor & HOD, ²B.Sc (CT) III Year

^{1,2}Department of Information and Computer Technology, Sri Krishna Adithya College of Arts and Science, Kovaipudur, Coimbatore, Tamil Nadu, India

How to cite this paper: Mrs. D. Seema Dev Aksatha | M. Blessing Marshal "Software Piracy Protection" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-3 | Issue-3, April 2019, pp.243-246, URL: <http://www.ijtsrd.com/papers/ijtsrd21705.pdf>



IJTSRD21705

ABSTRACT

Software piracy by users is generally believed to harm both software firms (through lower profits) and buying customers (through higher prices). Thus, it is thought that perfect and cost less technological protection would benefit both firms and consumers. The model developed here suggests that in some circumstances, even with significant piracy, not protecting can be the best policy, both raising firm profits and lowering selling prices. Key to the analysis is joining the presence of a positive network security with the fact that piracy increases the total number of program users. The network security exists because consumers have an incentive to economize on post-purchase learning and customization costs.

Copyright © 2019 by author(s) and International Journal of Trend in Scientific Research and Development Journal. This is an Open Access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (<http://creativecommons.org/licenses/by/4.0>)



Keywords: Copyright, Cryptography, Encryption, Authentication

INTRODUCTION

Software protection has recently attracted tremendous commercial interest from major software vendors to content providers including the movie and music recording industries. Their digital content is either at tremendous risk of arriving free on the desktops of millions of former paying customers or on the verge of driving even greater profit through new industries and new markets. The outcome may depend in large part on technical innovations in software protection, and related mechanisms for digital rights management(DRM) - controlling digital information once it resides on a platform beyond the control of the originator. Privacy advocates are interested in similar mechanisms for protecting personal information given to others in digital format. Related activities include Microsoft heavy investment in a next generation trusted hardware platform (Palladium) and the recent award by the US Air Force Research laboratory of us research contract involving software protection.

Software protection falls between the gaps of security, cryptography and engineering among other disciplines. Despite its name, software protection involves many assumptions related to hardware and other environmental aspects. A significant gulf currently exists between theory and practice. Inconsistencies have arisen in the relatively sparse open literature as a result of differences in objective, definitions and viewpoints. All of these issues provide research opportunities. We provide a selective survey on

software protection and related areas and encourage further research. We offer a number of viewpoints, discuss challenges, and suggest future directions. Software copy protection is a never-ending topic among developers. While it is true that perfect software copy protection is almost a dream given today's operating system and hardware infrastructure, if you are careful and use the right tools and techniques you can achieve a good degree of protection for your applications. This project is intended to maintain software copyright protection and assures that it is being accessed only by the authenticated users. Piracy has become so prevalent over the Internet that poses a major threat to ecommerce sites. With the help of malicious codes and programs, hackers or an intruder can gain access to the system and steal the information. Hence there arises a need to protect the information and products from being plagiarized.

This project is developed for the same purpose to protect the software's ownership of copyright and make transactions securely.

LITERATURE SURVEY

In this section we first discuss studies that analyze the economic impacts of software piracy. Behavioral studies that analyze demographic and cultural factors, and their effect on piracy behavior, are presented after the discussion of economic factors.

Economic Factors:

Economic models of piracy in general study the impact of piracy on profits and in particular the effect of enforcing copyright. Conventional wisdom suggests that piracy represents a drain to publisher profits and reducing piracy forces consumer to legitimately acquire software. We then identify various scenarios including the existence of a domestic software industry and study their effect on government incentive for increased copyright enforcement and publisher profits.

Behavioral and Ethical Factors:

Behavioral studies focusing on the demographics and personality characteristics of software pirates and the social and organizational context of software piracy are also gaining momentum. It is generally found that male pirates are more than females and that heavy users of personal computers pirate more software.

The importance of ethics in modeling software piracy is a recurring theme that is just beginning to be tapped. The decision to copy or not copy intellectual property is influenced by ethical reasons. Ethics is the study of moral systems. It is important to note that the moral philosophers does not make moral judgments about right or wrong but attempts to discover truth about the meanings of concepts and justification of judgments.

Consequentialism based theories examine ethical behavior in terms of the consequences or outcomes of activities. Consequentialism holds that people should identify the consequences of behaviors and evaluate the goodness or badness of all consequences.

PROPOSED SYSTEM

License management is the most important aspect of making money out of a software application. Only you or your company should be able to generate license keys, and you must be able to enforce using each license key on one computer (or a number of computers that you choose).

Many developers resort to hiding an obscure key generation and validation algorithm into the application, or encrypting a license key with a symmetrical encryption algorithm, and then hiding the encryption/decryption key into the application for license key decryption and validation. These methods are incorrect and weak, and if a malicious party is really interested in your application the decryption keys or decoding/encoding algorithm will be extracted from your application in a matter of days since release. This project is intended to maintain software copyright protection and assures that it is being accessed only by the authenticated users. Piracy has become so prevalent over the Internet that poses a major threat to e-commerce sites. With the help of malicious codes and programs, hackers or an intruder can gain access to the system and steal the information.

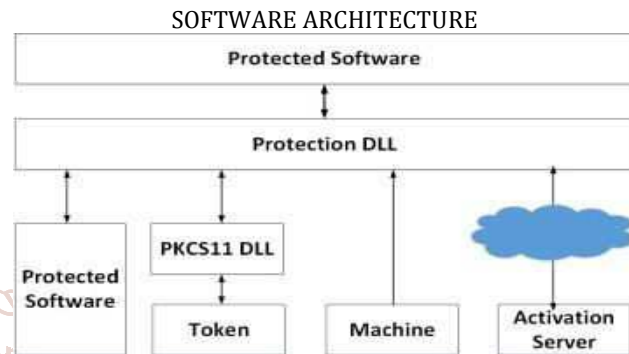
Hence there arises a need to protect the information and products from being plagiarized.

This project is developed for the same purpose to protect the software's ownership of copyright and make transactions securely.

ADVANTAGES:

- The system can be used by any online software dealer.

- It can also be used by organizations and business owners to promote their product and at the same time protect their copyright.
- If there would be no protection than anyone could copy and distribute the software and thus steal the copyright ownership.
- If the software gets pirated then it would lead to loss of revenue and can have a detrimental impact on the company thus this system is useful in making revenues.
- No one can copy the software or share as it requires an activation code that is different for each individual.
- Highly robust and secure system.

METHODOLOGY

The proposed SCP technique uses static and dynamic code injection and encryption to provide autonomous way for protecting the software code against expected attacks. The original software is modified to provide licensing and copy protection functionalities. illustrates the system proposed architecture and the main components:

Protected Software: A protected and modified version of the original software executable file/files.

Protection DLL: The core protection library responsible on providing different protection services and techniques.

License File: A secure file holding different rights and conditions applied to the protected software. **Activation**

Server: A web service responsible on providing the software activation.

Machine: The end user machine providing the Unique Serial Number used by Internet Activation technique.

PKCS11 DLL: A library responsible on managing the hardware token and providing the PKCS11 standard interface.

Token: A hardware component providing cryptographic and PKI services and a secure store for keys and Grants data.

IMPLEMENTATION**4.1 User Module**

The owner of the software generates the product key of the corresponding downloaded software and it's send to the user. After receive the key, when the user installs the software to the system it asks the product key, the user type that product key. It checks the product key is correct. If it is correct the software is installing, else the software is not installed to the system.

4.2 Enrollment module

An Enrollment module, the client first registers the all information (like name, address, username, and password) to the web server. After complete the registration process the client can be login to the system with the correct authentication.

4.3 Pay and Download Module

In this module is used the user view the software and then pay the amount of the software. The user download the software after he complete the buy process. After the download the admin generate the product key of the particular customer id. The admin send the product key to the correspondent user through SMS.

4.4 Activation Module

In this module is used to the user install the software to the system that requires the product key, after user enter the product key it checks if the entered product key matches the one which is provided by the code generator and then only allows to access the software product.

4.5 Admin Module:

4.5.1 Generate & Send Product key:

In this module, the admin generate the product key for correspondent customer id, and then send the product key to the particular user through SMS.

4.5.2 Verification Module

In this module is used to identify the authorized user. It is used to verify the entered product key and the product key generated from code generator. If it is mismatch, it does not allow the software install to the system.

LANGUAGE DESCRIPTION FRONT END TECHNOLOGY:- Microsoft .NET Framework:-

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet. The .NET Framework is designed to fulfill the following objectives:

To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet distributed, or executed remotely. To provide a code execution environment that minimizes software deployment and versioning conflicts. To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party. To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments. To Microsoft SQL Server is a Structured Query Language (SQL) based, client/server relational or indirectly in the completion of this work. make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications. To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.

The .NET Framework has two main components: the common language runtime and the .NET Framework class library. The common language runtime is the foundation of the .NET Framework. You can think of the runtime as an agent that manages code at execution time, providing core services such as memory management, thread management,

and remote, while also enforcing strict type safety and other forms of code accuracy that ensure security and robustness. In fact, the concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code. The class library, the other main component of the .NET Framework, is a comprehensive, object-oriented collection of reusable types that you can use to develop applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.

The .NET Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and unmanaged features. The .NET Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts.

BACK END TECHNOLOGY:-

About Microsoft SQL Server:- database. Each of these terms describes a fundamental part of the architecture of SQL Server.

Database

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

A database typically has two components: the files holding the physical database and the database management system (DBMS) software that applications use to access data. The DBMS is responsible for enforcing the database structure, including:

Maintaining the relationships between data in the database.

Ensuring that data is stored correctly and that the rules defining data relationships are not violated. Recovering all data to a point of known consistency in case of system failures.

CONCLUSION

This paper we presented techniques involved protection of software piracy, Which leads to access software only authorized person and prevent unauthorized access, This system provides more security of the software. Software piracy is a serious issue that impacts the bottom line for software developers. By implementing a security plan for software protection, software developers gain the benefits of protection from piracy as well as obtain the ability to implement additional license models. A security implementation plan that balances the time and resources with the desired outcome is possible given the wide range of security options. Developers can additionally choose a phased approach to the security implementation if time or resources are constrained in the short term.

Acknowledgment

I take the privilege to express my sincere thanks to **Dr. Rajendra Prasad**, Principal, Y.T.I.E.T, Bhipuri road and

Prof. Nilima Nikam, Head Department of Computer Engineering, **Yadavrao Tasgaonkar College of Engineering**, for providing the much needed support.

I wish to express my wholehearted appreciation and deep sense of heartfelt gratitude to my guide **Prof. Harish Barapatre**, Department Of Computer Engineering, Yadavrao Tasgaonkar College of engineering for her generous help, excellent guidance, lucid suggestions and encouragement throughout the course of this work. Her concrete directions and critical views have greatly helped me in successful completion of this work.

REFERENCES

- [1] J. Algesheimer, C. Cachin, J. Camenisch, G. Karjoth, "Cryptographic Security for Software", pp. 2-11 in Proc. 2011 IEEE Symposium on Security and Privacy (May 2011).
- [2] "The History of software piracy [infographic],"[online] 2011, <http://churchm/software-piracyhistory/>(Accessed: 18 October 2013).
- [3] "Digital Millennium Copyright Act,"[online] 2011, <http://whatis.techtarget.com/definition/Digital-Millennium-Copyright-Act-DMCA/> (Accessed:18 October 2013).
- [4] "Digital Millennium Copyright Act,"[online] 1998, <http://www.copyright.gov/legislation/hr2281.pdf>(Accessed:18October 2013).
- [5] <http://www.seopher.com/images/softwarepiracy-2007.jpg>
- [6] "Software Piracy: Study Claims 57 Percent of the World Pirates Software,"[online] 2012 ,http://www.huffingtonpost.com/2012/06/01/software-piracy-study-bsa_n_1563006.html (Accessed: 18 October 2013).
- [7] en.wikipedia.org.
- [8] <http://www.asp.net/>: This is the official Microsoft ASP.NET web site. It has a lot of: tutorials, training videos, and sample projects.

