



# International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991  
Vol. 1, No. 2  
April 2015



*2<sup>nd</sup> National Conference on "Recent Advances in Science  
Engineering & Technologies" RASET 2015*

*Organized by*

*Department of EEE, Jay Shriram College of Technology, Tirupur, Tamil Nadu, India.*



[www.ijerst.com](http://www.ijerst.com)

**Email:** [editorijerst@gmail.com](mailto:editorijerst@gmail.com) or [editor@ijerst.com](mailto:editor@ijerst.com)

*Research Paper*

# SOFTWARE LICENSE MANAGEMENT AND COPYRIGHT

**N Umanath<sup>1\*</sup>, M Vasanth<sup>1</sup> and N Thayani<sup>1</sup>**

*\*Corresponding Author: N Umanath*

The project aims at securing the software given to the customers such that the application runs only for particular count, up to a date, number of days starting from the installation date, runs in one machine. The application if copied to other system will not work. During first run, a license key is generated based on the Network MAC address which is unique in nature and the disk drive's serial number. The key is encrypted using Triple DES (Data Encrypted Standard) and stored in Windows Registry. In successive runs, the application looks up the value in the registry and may or may not continue. Thus the application aims in protecting the software piracy from one customer to another

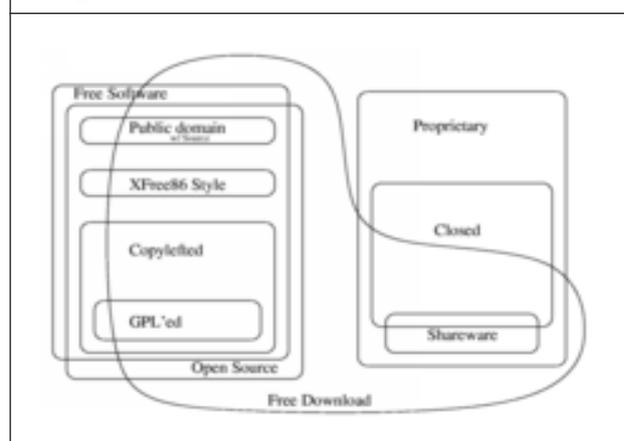
## PROJECT INTRODUCTION

The main objective of this project is securing the software given to the customers such that the application runs only for particular count, up to a date, number of days starting from the installation date, runs in one machine. The application if copied to other system will not work.

## SOFTWARE LICENSE

A software license is a legal instrument (usually by way of contract law, with or without printed material) governing the use or redistribution of software. Under United States copyright law all software is copyright protected, except material in the public domain. A typical software license grants an end-user permission to use one or more

**Diagram of software under various licenses**



copies of software in ways where such a use would otherwise potentially constitute copyright infringement of the software owner's exclusive rights under copyright law.

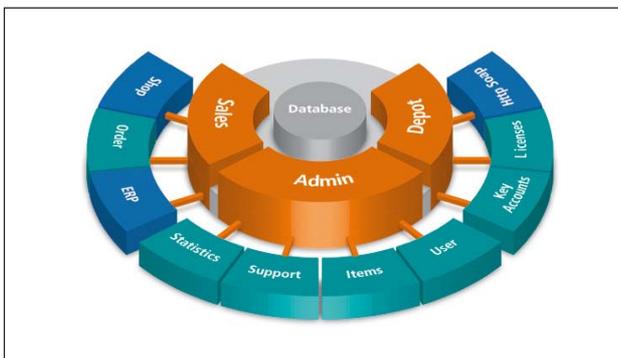
In addition to granting rights and imposing restrictions on the use of software, software licenses typically contain provisions which allocate liability and responsibility between the parties entering into the license agreement. In enterprise and commercial software transactions these terms often include limitations of liability, warranties and warranty disclaimers, and indemnity if the software infringes intellectual property rights of others.

Software licenses can generally be fit into the following categories: proprietary licenses and free and open source. The significant feature that distinguishes them are the terms which the end-user's might further distribute or copy the software.

### OVERVIEW

The project "Enhanced License Maker for Software Protection" is designed using Microsoft Visual Studio.Net 2003 as front end and Microsoft SQL Server 2000 as back end which works in .Net framework version 1.1. The coding language used is Visual Basic .Net.

The project aims at securing the software given to the customers such that the application runs only for particular count, up to a date, number of days starting from the installation date, runs in one machine. The application if copied to other system will not work.

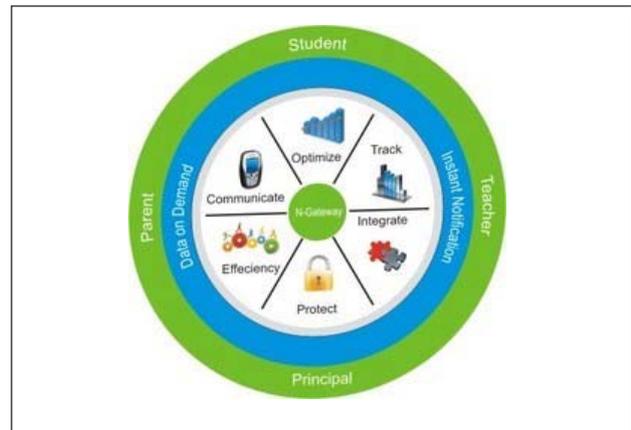


During first run, a license key is generated based on the Network MAC address which is unique in nature and the disk drive's serial number. The key is encrypted using Triple DES (Data Encrypted Standard) and stored in Windows Registry. In successive runs, the application looks up the value in the registry and may or may not continue.

Thus the application aims in protecting the software piracy from one customer to another.

### EXISTING SYSTEM

In Existing system, the software is not protected by giving proper security. Users can crack or hack the software and use the software unlimited no of times. Software Works by uninstalling the older or trial version or by editing the cleaning the registry.



### PROPOSED SYSTEM

In Proposed system, the software is fully protected by giving proper security permissions. Users can't crack or hack the software and use the software unlimited no of times. This is don't the editing values in registry and getting details about MAC Address, System Drive Information and etc. Software does not work by uninstalling the older or trial version or by editing and cleaning the registry

## Modules and its Description

There are 4 modules in this project. They are

1. Generate License
2. Getting MAC & Other Addresses
3. Reading & Writing Registry
4. Verifying Settings

### 1. Generate License

In the generate license module, we can give the installation date, number of days to be valid and end of day. The above details are to be stored in to database at the same time in database also. This information is checked at our protected software running.

### 2. Getting MAC & Other Address

In the second module, we are checking the MAC Address and IP Addresses for verifying system registry.

The MAC address is a unique value associated with a network adapter. MAC addresses are also known as **hardware** addresses or **physical** addresses. They uniquely identify an adapter on a LAN. MAC addresses are 12-digit hexadecimal numbers (48 bits in length).

By convention, MAC addresses are usually written in one of the following two formats:  
MM:MM:MM:SS:SS:SS

MM-MM-MM-SS-SS-SSThe first half of a MAC address contains the ID number of the adapter manufacturer. These IDs are regulated by an Internet standards body (see sidebar). The second half of a MAC address represents the serial number assigned to the adapter by the manufacturer. In the example, 00:A0:C9:14: C8:29 The prefix 00A0C9 indicates the manufacturer is Intel Corporation.

### 3. Reading & Writing Registry

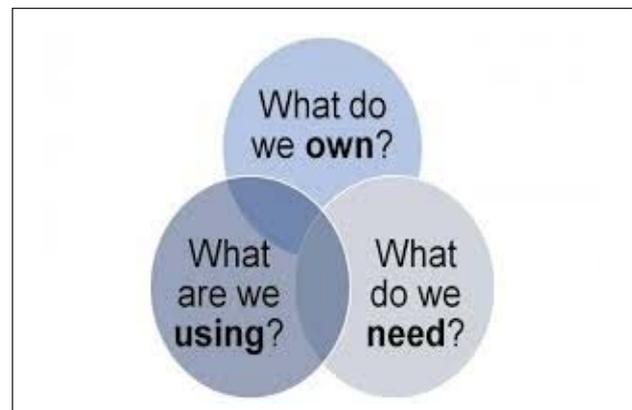
In the 3<sup>rd</sup> module we are accessing the system registry for getting the protection information. This information is verified at the time software running.

### 4. Verifying Setting

The Registry has a hierarchal structure, like the directories on your hard disk. Each branch (denoted by a folder icon in the Registry Editor, see below) is called a Key. Each key can contain other keys, as well as Values. Each value contains the actual information stored in the Registry. There are three types of values; String, Binary, and DWORD - the use of these depends upon the context.

### Security architecture/design analysis

Security architecture/design analysis verifies that the software design correctly implements security requirements. Generally speaking, there are four basic techniques that are used for security architecture/design analysis.



### Logic analysis

Logic analysis evaluates the equations, algorithms, and control logic of the software design.

### Data analysis

Data analysis evaluates the description and intended usage of each data item used in design

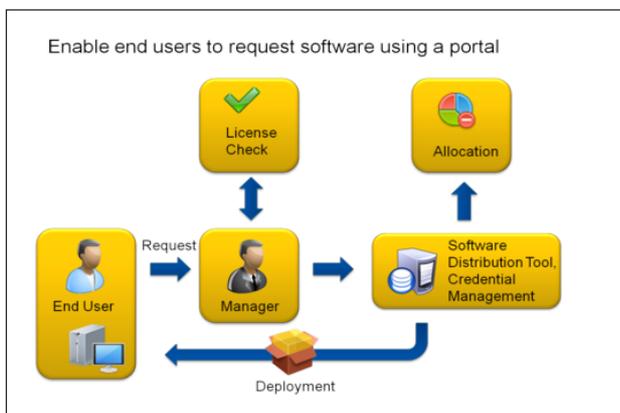
of the software component. The use of interrupts and their effect on data should receive special attention to ensure interrupt handling routines do not alter critical data used by other routines.

**Interface analysis**

Interface analysis verifies the proper design of a software component’s interfaces with other components of the system, including computer hardware, software, and end-users.

**Constraint analysis**

Constraint analysis evaluates the design of a software component against restrictions imposed by requirements and real-world limitations. The design must be responsive to all known or anticipated restrictions on the software component. These restrictions may include timing, sizing, and throughput constraints, input and output data limitations, equation and algorithm limitations, and other design limitations.



**Software licenses and copyright law**

In the United States, Section 117 of the Copyright Act gives the owner of a particular copy of software the explicit right to use the software with a computer, even if use of the software with a computer requires the making of incidental copies or adaptations (acts which could otherwise

potentially constitute copyright infringement). Therefore, the owner of a copy of computer software is legally entitled to use that copy of software. Hence, if the end-user of software is the owner of the respective copy, then the end-user may legally use the software without a license from the software publisher.

As many proprietary “licenses” only enumerate the rights that the user already has under 17 U.S.C. § 117, and yet proclaim to take rights away from the user, these contracts may lack consideration. Proprietary software licenses often proclaim to give software publishers more control over the way their software is used by keeping ownership of each copy of software with the software publisher. By doing so, Section 117 does not apply to the end-user and the software publisher may then compel the end-user to accept all of the terms of the license agreement, many of which may be more restrictive than copyright law alone. The form of the relationship determines if it is a lease or a purchase, for example *UMG v. Augusto*

**Hardware Requirements**

- SYSTEM : Pentium IV 2.4 GHz
- HARD DISK : 40 GB
- RAM : 512 MB

**Software Requirements**

- Operating system: Windows XP Professional
- Technology : Microsoft Visual Studio.Net 2008
- Coding Language : VB.Net
- Back End : SQL Server 2005



**International Journal of Engineering Research and Science & Technology**

**Hyderabad, INDIA. Ph: +91-09441351700, 09059645577**

**E-mail: editorijerst@gmail.com or editor@ijerst.com**

**Website: www.ijerst.com**

