

---

# RepublicanDRM – A Royalty Free License Manager User Guide & Product Overview

Dr Bradford Starkie

Starkie Enterprises, 52 Boisdale St. Surrey Hills, Melbourne Victoria,  
3127, Australia  
bstarkie@starkieenterprises.com  
<http://www.republicandrm.starkieenterprises.com>

**Abstract.** This document describes the Republican DRM license management system. It describes its features, and how to use it, in particular how to integrate it into your software, how to generate license keys.

## 1. Introduction

The Republican DRM License Manager is a software license manager that is royalty free.

- You pay the same low up-front fee (\$1000 AUD) regardless of how many copies of your product you sell.
- You do not pay any on-going charges.

What you get

- Java or C code to integrate with your product.
  - Standalone Software to generate license keys for your product.
  - PHP scripts to issue license keys to your customers, integrated with PayPal Instant Payment Notification.
- 
- Node locked, time based or feature based licenses or any combination of all three.
  - Unique 136 bit customer license keys generated by you for each of your customers to access your software.
  - Advanced anti-clock-tampering technology (only available on Windows 32 platforms)
  - 2 developer license keys (184-bit) to generate your licenses
  - Different keys for different products in your product range.

We begin this document with a description of the features of the product in sect

## 2. Features

Integrating RepublicanDRM with your Software

### 2.1. Anti-Clock Tampering Functions

One of the simplest ways in which users can circumvent some license managers is to roll their clock back when their license expires. To overcome this the RepublicanDRM system stores the time at which your application was last run in the Windows registry. Just after checking that your license is valid, the RepublicanDRM license manager checks to see if the application has been run at a point in time, after the current time. If so the license test fails. To recover from this all the user needs to do is to return their clock to the correct time, install a valid license key that they purchase from you, and the software will operate correctly.

Another way in which a user may attempt to circumvent license checking is to roll their system clock back prior to installing their license. If the license manager only

---

stored the expiry date of the license, this would mean that the license period would be greatly extended. To overcome these problems the RepublicanDRM license management system stores both a start date and an end date in the license key.

RepublicanDRM also stores in the registry the first time that the application is run. This enables RepublicanDRM to issue duration-based one-time only licenses, i.e. licenses that are valid only for a certain time after the application is first installed, for instance a 30 day once only trial.

The upside of using the registry to store time information is that it is more robust to attempts at license violations. For instance if the user deletes all traces of your application from their hard disk, there will still be a record in the registry of the first and last time that the application was run, making license violation more difficult. In addition the license keys are obfuscated (i.e. given unexpected names) making it harder for users to identify the registry keys, and delete or change them. The downside of using the registry is that advanced anti-clock-tampering methods are only available for Windows32 applications. In addition if your application is completely coded in Java you will still need to call a Windows executable to access the registry. This Windows application is included with RepublicanDRM.

## 2.2. Encryption Method

The Republican DRM algorithm uses a proprietary block cipher encryption algorithm. Table 1 below shows the statistics of this algorithm. The algorithm has the desirable property that each bit in the plain text is encrypted not just according to the encryption keys, but also according to every other bit in the plain text as well making it more robust to differential crypto-analysis.

Table 1 Statistics of the Encryption Algorithm

	RepublicanDRM
Block Size	136 bits
Mask Key Size	136 bits (customer specific)
Substitution Box	136 × 8 bits
Permutation	48 bits (customer specific)
Number of rounds	16

## 3. Integrating RepublicanDRM with your Code.

On 32 bit Windows operating systems you have the choice of using a Java API, C API or combined Java C API. On other operating systems only the Java API is available.

### 3.1. Java-only API

To use the Java Only API, all that is required is to use the **com.starkieenterprises.republican\_drm.LicenseManager** class. The **valid\_license** method is called and returns a true or false depending upon whether the application is valid. The customer's license key is stored in a file, in a location of your choice. The **valid\_license** method also needs your master key, which should be compiled into your application.

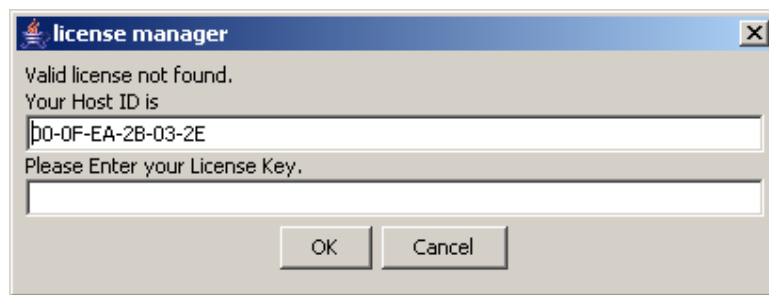
---

To make your task easier the Java **InvalidLicenseDialog** class is provided with RepublicanDRM. This dialog can be called when license management fails, so that installing a license key becomes a simple matter of your customers cutting and pasting the license key that you provide to them in an email or web form into the application. Your customers don't need to understand where in your source tree you expect them to store their license key. Advantageously the dialog also presents the node id of the computer in a text-field. Therefore if you decide to use node-locked licenses, users can cut and paste their host ID from this dialog and send it to you in an email, or paste it into a form on your website.

The RepublicanDRM classes are generic enough so that you can make the decision as to how error messages are presented, and what action is to be taken when license management fails. Depending upon your marketing strategy you may choose one of two options to take when license management fails as follows.

- You may choose to present the invalid license dialog to the user, such that the application will not run without a license or
- You may choose to run the application without a license, with limited functionality.

The choice is yours.



**Figure 1 The Invalid License Dialog included with RepublicanDRM**

See the JAVA API documentation [1] for more details, as well as the example application included with RepublicanDRM. The example applications are listed in the `Readme.txt` file in the root directory.

Note that if you want to use one-time licenses, or want to make use of the anti-clock-tapering technologies you will need to use the Java/C API listed below.

The Java API has one component that is OS specific, namely the collection of the node-id (MAC address) of the computer. This portion of the code has been tested on MAC OS 10 and well as Linux.

### 3.2. C API

Like the Java API, the API has a simple function call to implement the license test. Similarly a file is used, and is stored in a location of your choosing. Your master key also needs to be compiled into your application. There are two major differences between the Java API and the C API as follows.

There are no GUI classes, as this is less standardised  
Anti-clock-tampering and one-time licenses are provided

---

The library does include functions to test and store license keys, extract and present node id however. It is recommended that you use a similar strategy to that described in the Java API section of this document, namely to present the host ID to the user and to let them use cut and paste to store their license key, and for you to manage the saving of the key to a file for them.

To use one-time licenses, create a one-time license that operates on any host and distribute and install it with your application.

See the C API documentation [2] for more details, as well as the example application included with RepublicanDRM. The example applications are listed in the Readme.txt file in the root directory.

### 3.3. Combined Java/C API.

The Java API includes anti-clock-tampering and one-time licenses, while allowing you to code exclusively in Java. This configuration involves you distributing a C executable to do license checking which is called from an encapsulating Java Class. The interface between the C and Java components is encryption and makes use of challenge/response protocol, making it robust against substitution and playback attacks. You will need to compile your own version of this executable, and your master key needs to be compiled within it. We recommend that you use the MinGW G++ compiler [3] for this task., although the code will compile and link under Visual Studio version 6 if you prefer. The MinGW compiler and runtime environment and compiler is available from <http://www.mingw.org>

See both the JAVA API [1] and C API documentation [2] for more details, as well as the example application included with RepublicanDRM. The example applications are listed in the Readme.txt file in the root directory.

## 4. Generating License Keys

There are two ways for to create license keys, either

- Using the standalone key manager application or
- Using a PHP script on a website.

### 4.1. Using the standalone key manager application

Figure 2 below shows the key manager, If you have downloaded the Window 32 package then a shortcut to this application will exist on both your desktop, and on your program menu. The items should be intuitive. For instance to create a license that will run on any host, check the radio button next to the text “Run on any host”.

The key\_manager can be used to generate licenses for all of your different applications, regardless of whether the Java, C or Java/C API is used.

When the key manager is first installed, it is loaded with only one master key, which is the master key ...

```
C0189C6219D9C460D72434641CD3B4E9677B46200C9C993FED515578C695462
7C226326C229505A4312B29EA9B246483D81E9A08060C100005010D040A0B070
E0F020309
```

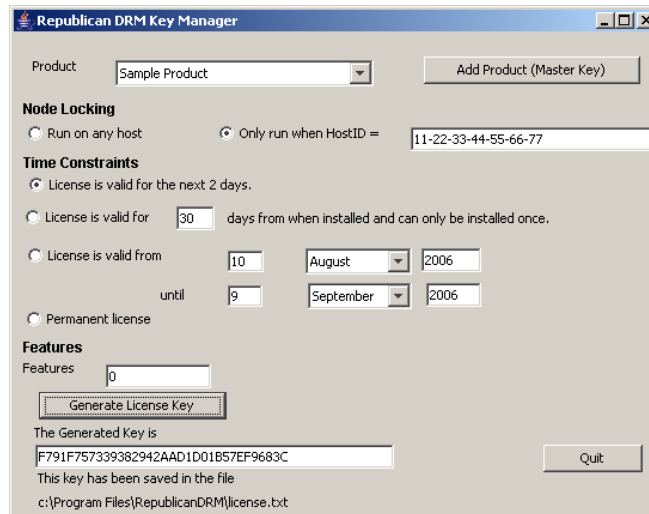
This key can only be used to create fixed two-day licenses. To generate other kinds of license keys you need to purchase a license from

[www.republicandrm.starkieenterprises.com](http://www.republicandrm.starkieenterprises.com). To load new keys into the key manager,

---

click on the “Add Product” and you will be presented with a dialog into which you can paste your license key. Otherwise you select the product you wish to create a license for from the dropdown box, and then fill in the remaining details.

Once you have filled in all of the details click on “Generate License Key” and the new license key will be presented in the text box at the bottom of the screen. The license is displayed in a text box, so that cut and paste will work on it. The key will also be saved in a file. The location of the file is display at the bottom of the screen. If the path name of this file is so long that the end of the text is not displayed, simply place your cursor over the text, and the full pathname will be displayed in the tool tip text.



**Figure 2 The Republican DRM Key Manager**

#### 4.2. Using a PHP script on a website.

There are two PHP scripts included in the PHP tarball for generating license keys. The first of these is admin/mkkey.html and admin/mklicensekey.php. This pair of scripts is an online equivalent of the key\_manager, which can be stored in a PIN protected directory on the server. In fact the layout of the form is identical to the layout of the Key Manager.

The second of these is the register.html and getlicensekey.php scripts. These scripts are integrated with online purchasing (PayPal Instant Payment Notification), email and MySQL. The scripts are encapsulated well enough that you should be able to modify them to integrate with other payment mechanisms and other HTML scripts. The business logic behind these scripts is that a license key will only be generated if there is an entry in the database for a given email address that states that payment has been received for that customer. When the key is generated, an email is sent to the given email address containing the license key. Please see the PHP API reference guide for more details. [4]. You can find a complete working version of the PHP scripts at [http://www.hello\\_world.starkieenterprises.com](http://www.hello_world.starkieenterprises.com). This site is fully operational except that it does not transfer any money. You can simulate purchasing of products and issuing of licenses on this site.

### 5. Additional Information

---

If you have any more questions please contact  
<mailto:support@starkieenterprises.com>.

## 6. References

1. Starkie, B., *Republican JAVA API documentation*.  
2006[http://republicandrm.starkieenterprises.com/java\\_docs/](http://republicandrm.starkieenterprises.com/java_docs/).
2. Starkie, B., *RepublicanDRM C API Reference*.  
2006[http://www.republicandrm.starkieenterprises.com/c\\_doc.htm](http://www.republicandrm.starkieenterprises.com/c_doc.htm).
3. [www.mingw.org](http://www.mingw.org), *MinGW Minimalist GNU for Windows*.  
2006<http://www.mingw.org>.
4. Starkie, B., *RepublicanDRM PHP Scripts*.  
2006[http://www.republicandrm.starkieenterprises.com/php\\_documentation.htm](http://www.republicandrm.starkieenterprises.com/php_documentation.htm).