Persistent Security

SafeGuard LM 5.6 Anti-hacking module
(25-apr-2016)

Welcome to SafeGuard LM 5. SafeGuardLM is comprehensive software license management for small to large businesses. Whether you are a small developer working with C/C++, Java, Python, Objective C applications, PHP 5, or a business that has thousands of users, we provide the optimal solution to protect your software.

Although, we have not had any reports of compromised applications at this point, we continually strive to provide every possible means to further the protection of your software. We have come up with a simple yet effective method to increase the level of protection against hackers. This document and software is made available only to paid customers and is not part of the Evaluation version of SafeGuard LM. Neither is the psseed utility that is used to plant your private encryption seeds into libraries and applications.

The "*secure*" folder contains this document along with object code and source code to further enhance security in your applications on Windows, OS X and Linux platforms. This anti-hacking module is ONLY available to C, C++ and Objective C applications, as you must link in the special object file when building your application.

By default, all you need to do is link the smonitor object file into your applications main(). The GNU compiler or Windows runtime will launch the thread automatically. The default parameters allow for the application to take up to ten (10) seconds to make a call to one of the following SafeGuard LM functions that either can return a success or failure return code to at least one of the following function calls.

sgAuthorized()
sgCheckout()
sgPaActivate()
sgPaGetDemoActivationKeys()
sgPaGetPaidActivationKeys()

If after that ten (10) seconds none of these calls have been made, the application WILL terminate unexpectedly.
This is to combat what hackers do with software like IDAPro to "remove" or "jump" over function calls in the applications binary executable by systematically altering the assembly code in the executable to circumvent license checks.

We are providing you (a paid customer) with the source code of this module. The reason being is that it would allow you to alter the default ten seconds until termination and/or the function names that perform this work. The existing function names are fairly obscure. You may want to make them unique to your company.

The code actually starts up a "monitor" thread when the application is launched. This is done without you needing to perform any work or function calls yourself. This thread is then notified by the internal license code that the application either performed one of the above function calls or not. If the monitor thread is not notified in the time allotted, the application will terminate. If for some reason, a hacker spent enough time to find and disable the exit()/ExitProcess() calls in the monitor thread, the application will then terminate ungracefully with a Segmentation Fault or Access Violation.

If before the default ten (10) seconds are up, the monitor thread is notified that everything is OK, the monitor thread will simply kill itself as no more monitoring is necessary and your application will carry on.

If for some reason you have an application that does not normally stay running for more than ten (10) seconds and you still wish to further enhance the protection, you can make a function call sometime after the license check is performed and this special call will perform the hacking check immediately and ether terminate the application or let it continue.

This function call is sg_initdevice(“some string”); The string content is irrelevant. It will instruct the monitor thread to immediately check whether a license check has already been performed or not and take the appropriate action.

We hope you find this security enhancement to your liking and if you have any questions or comments, please contact us.

Best regards, and safe computing,

Thank you,