Protection and Licensing

Guardant makes it possible to create protection systems of any level of complexity. Guardant software provides effective tools for ensuring protection of any software product.

The tamperproofness and reliability of protection directly depend on the level of elaboration and correctness of protection system implementation. Here we describe the main actions that you need to perform in order to properly apply a protection system.

Dongle-based protection systems can perform a range of checks of various degree of complexity. The most elementary are verifications of the dongle's presence with a range of preset attributes. These can be performed fast and rather frequently.

More complex checks use the encryption of the information with the use of dongles. Since a dongle is an intelligent device, it can perform the encryption of information using special algorithms.

Developers of a protection system can create algorithm descriptors and record those into dongle memory by themselves using the programming utilities (except for Guardant Fidus dongles). This is what makes every protection system using algorithms unique.

Guardant software supports two methods of protection:

- Automatic protection of executables
- Protection using Guardant API

Protection Principle

Software/hardware Guardant dongle serve for protecting software against computer piracy: illegal copying, distribution and use of applications.

The protection system is based on using a dongle. Dongle is a small electronic device, which connects to a computer. Protected applications are 'bound' to the information contained in a dongle.

blocked URL

In general terms the system's principle of operation looks as follows:

- The protected application accesses the dongle, which should be connected to the computer
- The dongle returns some information to the application
- The application identifies the dongle using this information. If the dongle contains valid parameters, the application continues to operate. If the dongle's parameters are invalid (not the right dongle) or the dongle is missing, the protected software application halts

blocked URL

Automatic protection

This method is based on the processing of executable files (Win PE) by the Guardant automatic protection utility. As a result, the application gets tied to the dongle and gains protection against debuggers and disassemblers. The automatic protection has a range of options (modes), which serve for adjusting an application for the dongle parameters (tie it to the ID, serial number, etc.), limiting the number of launches or running time of the application, encrypting it. The option of time-to-time dongle verification is also supported.

The main advantage of this method is the short timeframe that it requires for setting up the protection. This process takes just a few minutes. Besides, this does not require any special skills. This method is useful even when the source code of the application is missing and the development of a protection system based on Guardant API is impossible.

The main disadvantage of this method is that it cannot provide a sufficient level of protection to the application. The essence of the method indicates that the protection is not completely integrated into the application. It 'wraps' an executable file, as though it is 'glued' to it, therefore, there is a chance that it can be broken by a hacker. Besides, this method cannot provide for any dedicated logic of the protection system operation, which is very important in increasing the protection system tamperproofness.

Automatic protection operation scheme:

- The automatic protection utility embeds an executable module into the body of protected application – internal vaccine.
- At the moment of launching the application the internal vaccine calls the external vaccine from a separate file.

The external vaccine runs the required checks and encryptions and launches the protected application.

Guardant API

This method is based on the use of special Guardant API functions located in object modules. API functions provide for performing any operations with a dongle: search, reading and writing into is memory, setting of hardware locks, encryption of data using the hardware algorithms, etc. To setup protection using this method you need to embed calls of API functions into the source code of the application and compile them with the object modules.

The main advantage of this method is that it provides an immeasurably higher level of protection. The protection (if implemented correctly) becomes integrated into the application, which makes it hard to remove by a hacker. Guardant API functions serve for performing any operation with a dongle, they can process any accessible area of its memory – in other words, the possibilities of developing protection are limited only by the developer's fantasy and abilities. You can build any, even highly dedicated logic of protection in order to significantly complicate the hacker's task in breaking it. Finally, only Guardant API functions provide a complete freedom of actions while working with dongle hardware algorithms.

The creating of protection build on Guardant API is a task assuming a range of various solutions. Therefore, it is impossible to offer a universal and detailed step-by-step process description for such protection. Below you will find a generic action plan that you should be guided by in any case:

- Study the test examples of programs written in respective program-ming language (see Folder **"C:\Program Files\Guardant\SDK7\Samples"**). The tests contain examples of using the main API functions
- Develop you own protection system using the acquired knowledge and recommendations described in Appendix on **Increasing Tamperproofness of Protection**

What method to choose?

Of course, you can use these methods separately – only the automatic protection or only the API-based protection. However, all of the above leads to the conclusion that both methods should be used together. This is the only way to combine the advantages of both methods and mitigate their disadvantages. Use the automatic protection to protect an application against debuggers and disassemblers, encrypt its body, and conceal the calls of API functions from the outsiders. Besides, this is a great method of protection from curious users not sufficiently skilled to break the protection system.

However, the automatic protection should remain just the outer defense position. At its core the protection must be based on the use of API functions. That is where the main work functions must be performed: dongle presence verification and reaction to missing dongle, operations with memory and hardware algorithms, etc. The most important thing is to build protection so that it becomes an integral part of protected application that it will require in order to operate properly.

Follow the main rules listed below while organizing a protection system. These rules will help you build a more effective and reliable protection.

- Combine the automatic protection with protection based on the use of API functions
- Use hardware data encryption algorithms
- In case of automatic protection of applications use the following options wherever possible:
    - Encryption of the loadable part of application
    - Regular verification of dongle presence
- In case of APIbased protection:
    - Do not keep the access codes in the application body in an explicit form
    - Use complex algorithms working with API functions
    - Distribute checks about the application code
    - Use various checks with different expectancy
    - Delay the application reaction to return codes of API functions
    - Complicate the logic of processing the return codes