Protection techniques provided by hardware algorithms

It is important to be creative while using the hardware algorithms. Indeed, you have a wide range of ways for converting information pursuing different goals. The tamperproofness of protection directly depends on how original and unusual methods of using hardware algorithms and converted data by the protection system are.

Several Hardware Algorithms in One Dongle

You can create several descriptors of hardware algorithms defining the ways for data conversion into a single Guardant dongle. At any time you can use any of the activated descriptors of hardware algorithms. You can use all descriptors created in the dongle for protecting the same application: for example, one part of the data may be encrypted with one algorithm, and another – with another one, etc.

Using **GrdUtil.exe** dongle programming utility you can easily build determinants of hardware algorithms and define their properties. You can also develop your own alternative utilities performing similar functions but programming the dongles using the technology implemented by developers for a specific company.

If a company produces several software products, you can and you should protect every one of them using unique descriptors of hardware algorithms. It is also possible to have unique descriptors in each dongle. This will provide a unique protection method for each copy of the application. Using this feature of Guardant dongles eliminates the danger of creating universal emulators for various software products (or for different versions of one product).

The possibility of activating and deactivating hardware algorithms provides for writing their descriptors for the future use to be further used in new versions of the application. These algorithms may remain in deactivated status prior to the new version release and get activated during the upgrade to the new version.

Random numbers and analysis of probability function

You can build reliable protection schemes based on the probability principle. For example, we create an algorithm which returns a random sequence of the 100 bytes size. It is divided to 50 numbers, 2 bytes each. We calculate the expectation function, mean square deviation and check if the probability function has uniform distribution. If it is not so, that means that the hacker is trying to emulate the protection against copying, since it is hard for him (and nowhere from) to take that many correct random sequences with uniform distribution. If the application periodically accesses the dongle, the statistics can be accumulated and refined on every call giving more and more accurate statistical picture.

Thus, in order to hack the protection, a hacker needs to write a rather bulky program with no clear vision of how to embed it into the protected application. Or he will have to analyze the entire system, which can take a lot of time if the system of collecting and analyzing the statistics is spread throughout the application.