Native Executables Protection

Guardant dongle is a highly effective mean of software/hardware protection. It allows building protection of virtually any level of complexity and tamperproofness.

Use automatic protection utilities for protecting ready-to-use applications against the attempts to learn their operating logic. To ensure proper protection it is recommended to use the following options:

- **Protection against viruses**. This option is used to protect an application against illegal modification of the application file (incorporating program modules, alteration of copyright information, etc.).
- **Encryption of the loadable part and internal overlays of the application**. This option protects the application against attempts to use another method of code research – disassembling.
- Use the **time-dependant dongles checks** options. In this case even full dump of the application memory will be just a waste of time for a hacker.
- Use **import and code extraction protection** options. These options allow the autoprotection to extract a range of instructions from the body of the protected application and transfer it into the virtual machine's body. This allows withstanding the automatic deactivation of hinged protections which significantly increases the difficulty and cost of attack.

Important information

If the use of **/RIP_CODE** and **/IMPORT_HOOK** options significantly slows down the work of the application, it is recommended to use **/RIP_CODE_LIST** and **/IMPORT_HOOK_LIST** options for optimizing the speed of the protected application.

> In the event when the tie is done to Sign or Time dongle use **ECC160 asymmetric algorithm operating option**. Whereas, random data will be generated and signed by a digital signature on elliptic curves directly in the dongle in the course of operating the protected application. Later the signature will be verified by Guardant API function, protected by pseudo code, traffic encryption and other protection mechanisms.

Autoprotection Principle

The Native protection is based on a vaccine implemented as a universal external module. All protection functions are supported by this module.

This serves for complete unification of the protection process.

Here is the description of Guardant automatic protection:

A small executable module is embedded into the body of the protected application (internal vaccine). At the application startup it loads an external vaccine from a separate file. And the latter performs all necessary checks and conversions of protected application's code and launches of the application.

Besides, the use of the external module of protection enforces the tamperproofness of the system and prevents the learning of its logic using debuggers.

The Guardant vaccine file is named **GrdVkc32.dll,** which is included into the automatic protection kit.

Warning

GrdVkc32.dll vaccine must be located in a folder accessible by **LoadLibrary** function during the protected application startup (in the current folder, system folders of Windows, protected application folder, one of the folders of PATH list).

Native Autoprotection Limitations

Important information

1. Autoprotection should be set using a dongle of the same model that will be shipped with the protected application.
2. To ensure the successful setting process of autoprotection as well as protected application running, the dongle [to which the application is being bound] must contain symmetric-key algorithm (GSII64 or AES).
3. The hardware algorithm determinant in the dongle used for protection should be identical to the determinant of the same algorithm in the dongle from the protected application package.

- The utility does not support selfextracting archives, ZIP, RAR, etc
- The autoprotection utility does not support installation wizards created in special development environments: Wise Installer, Install Shield, etc
- We cannot guarantee proper protection and correct operation of applications previously packed by a special archiver of EXE files: UPX, ASPACK, etc
- We also cannot guarantee proper protection of EXE files with previously protected code against modifications or analysis.

More information:

- Native Autoprotection Console Utility
  - Files required for the protection process
  - Files required for the protected application
  - Native autoprotection procedure
  - Error Codes
  - Brief Description of Protection Options
- Automatic Protection Options
  - Dongle Type Setting Options
    - Set binding to dongle type
  - Dongle Binding Options
    - Dongle ID verification
    - Dongle serial number verification
    - Protected application version verification
    - Mask verification
    - Program number verification
    - Restrict the use of hardware algorithms
    - Control USB Dongle's Presence
    - Verify the dongle presence periodically
    - Display 'dongle missing' message a preset number of times
    - Delay application shutdown
  - Options increasing application protection
    - Cancel application encryption
    - Turn off encryption of initialized data
    - Integrity verification
    - Restrict application 'stripping'
    - Change the number of algorithm queries
    - Pack executable file sections
    - Control memory page attributes
    - Implicit linking support
    - Protect imported functions
    - Protect imported functions using a list
    - Extract instructions from the body of the application
    - Extract instructions from the application's body using a list
    - Verify the digital signature
  - Network Protection Options
    - Select the licensing mode
    - Set the number of licenses for multi-module network application
  - Licensing options
    - Show an expiry date warning
    - Restrict the number of application launches
    - Show a warning on the remaining number of launches
    - Show a link to a developer's website
  - Service Protection Options
    - Creation of a customized file with vaccine messages
    - Show the splash screen at the startup
    - Defining output path
    - Restriction of displaying utility messages on the screen
- Defining List of Files for Protection
- Use of .FIL List File