Автозащита Native-приложений

Для защиты готовых приложений от изучения логики их работы пользуйтесь утилитой автоматической защиты. При этом целесообразно использовать следующие опции:

- **Защита от вирусов**. Эта опция позволяет защитить приложение не только от файловых вирусов, но и от нелегальной модификации файла приложения (дописывания каких-то программных модулей, изменения копирайтов и т. п.).
- **Кодирование загружаемой части и внутренних оверлеев приложения**. Это предохранит приложение от попыток использования другого метода изучения – дизассемблирования.
- Используйте опции **проверки электронных ключей по времени**. В этом случае даже снятие полного дампа памяти приложения окажется для хакера пустым развлечением.
- Используйте опции **защиты импортов и извлечения кода**. Эти опции позволяют автозащите извлечь ряд инструкций из тела защищаемого приложения и перенести их в тело виртуальной машины. Это позволит противостоять технологиям автоматической деактивации навесных защит, что существенно увеличит сложность и стоимость взлома.

  Важная информация
  Если применение опций /RIP_CODE и /IMPORT_HOOK сильно замедляют работу приложения, рекомендуется использовать опции /RIP_CODE_LIST и /IMPORT_HOOK_LIST для оптимизации быстродействия защищеной программы. Подробнее см. Руководство пользователя, Часть 1, глава Автоматическая защита

- В случае, когда осуществляется привязка к Sign или Time-ключу, используйте **опцию работы с ассиметричным алгоритмом ECC160**. При этом в ходе работы защищенного приложения будут генерироваться случайные данные и подписываться цифровой подписью на эллиптических кривых непосредственно на электронном ключе. Затем подпись будет проверяться функцией Guardant API, защищенной псевдовокодом, шифрованием трафика и другими защитными механизмами.