

Символьная обфускация .NET-сборок

Символьная обфускация позволяет усложнить жизнь злоумышленнику, так как все осмысленные имена внутри защищаемого приложения превращаются в набор случайных символов, и обратный анализ такого кода становится гораздо сложнее. Если осмысленные имена больше не используются, то поиск по ключевым словам становится невозможным. Например, «`SerialNumberCheck(uint licenseID, byte[] serialNumber)`» в виде обфусцированного имени будет выглядеть, как `"abb(uint zxd, byte[] skf)"`.

Технология перегрузки имен, также использованная в обфускаторе, позволяет присвоить одно и то же имя большому количеству сущностей внутри защищаемого приложения, что позволяет еще сильнее усложнить его обратный анализ.

Еще одним элементом защиты, который реализуется посредством обфускации, является шифрование строковых констант с использованием аппаратного алгоритма электронного ключа. Этот механизм позволяет обращаться к ключу в процессе выполнения, причем для каждого защищаемого приложения такая схема работы с ним будет уникальной, так как обращение к строковым константам происходит в разных местах обфусцированного кода.

Обфускатор Guardant предоставляет следующие возможности для защиты .NET-приложений:

- Обфускация исполняемых файлов (.exe) и библиотек (.dll)
- Символьная обфускация имен (типы, методы, поля, свойства, события, параметры методов и обобщений)
- Перегрузка полей, методов (присваивание одного и того же имени объектам с разными сигнатурами)
- Поддержка нескольких алфавитов (символы, цифры, символы и цифры, непечатные символы)
- Тонкая настройка параметров обфускации при помощи конфигурационного файла
- Сохранение в файл отображения обфусцированных имен в оригинальные
- Сохранение статистики обфускации (покрытие оригинального кода обфусцированным)
- Шифрование строк с использованием электронного ключа
- Обфускация графа потока управления

- [Обфускация](#)
- [Файлы, необходимые для процесса защиты](#)
- [Файлы, необходимые для работы обфусцированной .NET-сборки](#)
- [Порядок защиты](#)
- [Общие и индивидуальные опции обфускации](#)
- [Сводные таблицы опций обфускатора](#)
- [Опции установки типа электронного ключа](#)
 - [Привязать к типу ключа](#)
- [Опции привязки .NET-сборки к электронному ключу](#)
 - [Проверять ID ключа](#)
 - [Проверять версию защищенного приложения](#)
 - [Проверять серийный номер ключа](#)
 - [Проверять маску](#)
 - [Проверять номер программы](#)
 - [Выводить сообщение об отсутствии ключа заданное число раз](#)
- [Опции обфускации .NET-сборки](#)
 - [Выполнить символьную обфускацию .NET-сборки](#)
 - [Использовать результаты работы Exclusion Utility](#)
 - [Зашифровать строковые константы .NET-сборки](#)
 - [Обфусцировать публичные интерфейсы .NET-сборки](#)
 - [Создать защищенное хранилище](#)
 - [Генерировать исключение при проблемах с ключом](#)
 - [Генерировать файл результатов обфускации](#)
 - [Ускорить генерацию таблиц вопросов в процессе защиты](#)
 - [Задать процент обфускации графа потока управления](#)
- [Опции, ограничения времени работы и числа запусков приложения](#)
 - [Вывести предупреждение об оставшемся времени использования](#)
 - [Выдать ссылку на сайт разработчика приложения](#)
 - [Ограничение количества запусков приложения](#)
 - [Отобразить предупреждение об оставшемся числе запусков](#)
- [Сетевые опции](#)
 - [Выбрать режим лицензирования](#)
 - [Использовать систему управления лицензиями](#)
- [Опции, влияющие на защищенность приложения](#)
 - [Задать число ключей шифрования](#)
 - [Задать число таблиц вопросов-ответов к алгоритму](#)
- [Сервисные опции автозащиты .NET-сборки](#)
 - [Задать выходной путь](#)

- Задать собственные сообщения вакцины
- Запретить вывод сообщений защищенного приложения
- Запретить вывод сообщений утилиты на экран