Guardant®

Система защиты от компьютерного пиратства

Эффективная
защита приложений

Вводный урок:
Выбор модели ключа,
построение проекта защиты

Содержание

Введение

Подъем индустрии программного обеспечения продолжается: появляются новые продукты, растут объемы продаж, а значит — и доходы разработчиков. В этих условиях высок риск потерять часть прибыли за счет брешей в защите продукции. Прикрыть кормушку исключительно правовыми средствами пока нереально, поэтому проблема защиты ПО от хакеров и пиратов всех мастей стоит на данный момент достаточно остро.

Эта серия уроков научит реализовать качественную защиту приложений с использованием технологий Guardant.

Рамки данного занятия охватывают общее описание процесса проектирования и реализации защиты, а также возможности, предоставляемые комплектом разработчика и электронными ключами Guardant.

Используемые термины и обозначения

- **Лицензирование ПО**: условия предоставления прав на использование ПО конечным пользователям.
- **Лицензионная политика**: порядок предоставления, изменения и отзыва этих прав.
- **Защита**: комплекс организационных, юридических и технических мер для реализации лицензионной политики. (Далее в контексте уроков защиты рассматривается комплекс технических мер).

Анализ объекта защиты

Прежде всего, необходимо определиться со рядом вопросов, от которых зависят требования, предъявляемые к защите требования:

- **Условия эксплуатации приложения** (ориентировано ли оно на работу в сети, или предназначено для исполнения локально);
- **Используемые механизмы предоставления, изменения и отзыва прав** (разовая продажа лицензий или дополнительные возможности, такие как ограничение действия лицензий по астрономическому времени, удаленное управление предоставленными привилегиями и т. п.);
- **Доступность исходного кода** защищаемого приложения,
- **Наличие участков кода**, изучение которых желательно предотвратить

Формирование защиты приложения

После того как условия эксплуатации защищенного приложения, а вместе с ними особенности лицензионной политики, определены, можно приступать к реализации защиты.

Предлагается следующий алгоритм действий (см. схему 1):

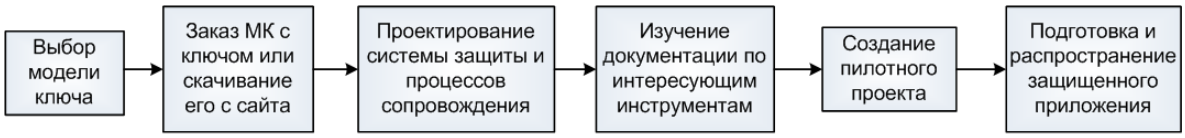

Схема 1. Алгоритм формирования защиты

Остановимся подробнее на каждом из этапов.

Выбор модели ключа

Выбор модели ключа основывается, по большей части на особенностях предполагаемой лицензионной политики и способе эксплуатации защищаемого приложения.

В частности, если приложение предназначено для работы в локальной сети, то лицензирование копий приложения можно осуществлять по сети. Это может быть реализовано на основе сетевых ключей – **Guardant Sign Net** и **Guardant Time Net**.

При необходимости ограничения работы приложения по времени имеет смысл использовать ключ **Guardant Time**. С его помощью можно организовать активацию/деактивацию функционала защищаемого приложения по времени.

Потенциально может возникнуть необходимость надежной защиты некоторого алгоритма обработки данных от копирования. Такая возможность предоставляется ключом серии **Guardant Code**, на основе которого можно создавать уникальные схемы защиты с выполнением загружаемого кода непосредственно внутри ключа.

Проиллюстрируем сказанное схемой 2:

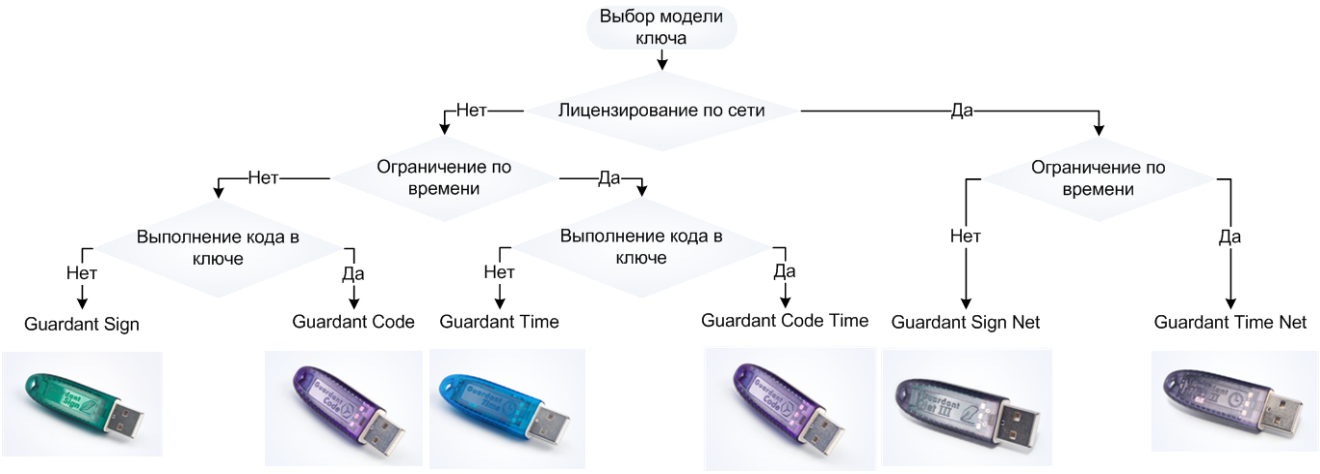

Схема 2. Выбор модели ключа

Как только выбор подходящей модели ключа сделан, можно заказать ключ, а при необходимости, и коробочную версию комплекта разработчика.

Примечание

Существует возможность получить под залог стоимости полнофункциональный ключ любой модели с общеизвестными кодами доступа, который в течение трех месяцев можно вернуть, если по какой-то причине он не подойдет, или обменять на «боевой» ключ той же модели.

Проектирование защиты и процессов сопровождения приложения

Обратимся к вопросу, заданному в начале урока: доступен ли исходный код защищаемого приложения?

Если приложение, подлежащее защите, доступно только в виде исполняемого файла, то защитить его можно с использованием ***Автозащиты Guardant***. Однако при наличии исходного кода, а также ресурсов на разработку и внедрение собственной системы защиты при помощи ***Guardant API***, рекомендуется поступить именно так (автозащиту в этом случае можно и нужно использовать в качестве механизма, усиливающего стойкость приложения к анализу и несанкционированной модификации).

Преимущество создания собственных уникальных механизмов защиты на основе функций API очевидно: только в этом случае стойкость создаваемой защиты будет напрямую зависеть от мастерства и фантазии ее разработчика.

Подробнее принципы реализации защиты при помощи Guardant API будут рассмотрены в одном из следующих уроков. Также множество полезной информации можно почерпнуть в ***Документации Guardant***.

Так как проектирование любой системы начинается после исследования возможностей доступного инструментария, то в рамках данного урока имеет смысл перечислить некоторые из инструментов защиты и механизмы реализации лицензионной политики, предоставляемые комплектом разработчика и электронными ключами Guardant.

Состав Комплекта разработчика (называемого также Мастер комплект, МК) можно представить следующим образом:

Пиктограмма	Название	Назначение
	Интегратор	Обеспечивает быстрый доступ ко всем утилитам МК
	Автозащита	Набор утилит для автоматической защиты исполняемых файлов, включая графические и консольные инструменты

Пиктограмма	Название	Назначение
	Редактор памяти ключа	Утилита ***GrdUtil***, предоставляющая разработчику защиты возможности как локального, так и удаленного редактирования памяти ключей, а также работы с БД информации о клиентах
	Утилиты для удаленного обновления ключей	Утилиты ***GrdTRU*** и ***GsRemote***, передаваемые конечному пользователю для осуществления разработчиком процедур удаленного обновления памяти ключа
	Утилиты для работы с сетевыми ключами	Сервер ключей ***GrdSrv***, предоставляющий доступ к сетевым ключам Guardant в пределах локальной сети и утилита сопровождения ***GrdMon*** для удаленного мониторинга корректности работы сервера
	Утилита диагностики	Утилиты ***GrdDiag*** и ***GrdDem32*** для диагностики и устранения неполадок, а также для сбора и отправки сведений о системе и имеющихся ключах в службу технической поддержки
	Драйверы	Комплект ***драйверов*** для работы с ключами (существует также возможность работы ***без драйвера***, т.е. в режиме ***HID***)
	Guardant API	Библиотеки и объектные файлы, необходимые при реализации защиты с использованием ***Guardant API*** на одном из следующих языков: ***C/C++***, ***C#***, ***Delphi***, ***Java***, ***Visual Basic***
	Документация	Комплект документации

Электронные ключи Guardant - это устройства, способные обрабатывать информацию в соответствии с реализованными в них криптографическими алгоритмами и обладающие защищенной от несанкционированного считывания памятью для хранения информации, используемой для защиты приложений. В зависимости от модели выбранного ключа, предоставляются следующие возможности:

Guardant Sign

- Реализация симметричных и асимметричных криптографических алгоритмов внутри ключа;
- Хранение секретных ключей криптографических алгоритмов и их использование при выполнении криптографических операций только внутри ключа;
- Хранение любой пользовательской информации, необходимой для защиты приложения, в защищенных ячейках ключа (с возможностями ее считывания и изменения по паролю);
- Возможность удаленного обновления содержимого защищенной области памяти ключа, находящегося у конечного пользователя защищенного приложения, включая параметры и ограничения работы записанных алгоритмов.

Guardant Time

- Поддержка функциональности Guardant Sign;
- Возможности ограничения работы алгоритмов по времени (предоставляются сервисы активации/деактивации алгоритмов – как в определенный момент времени, так и по прошествии определенного времени после первого обращения к алгоритмы);
- Возможность автоматического изменения ключей шифрования алгоритмов с течением времени (может быть использовано, к примеру, для прогнозируемого периодического изменения ключа шифрования, используемого для защиты информации, передаваемой между клиентом и сервером).

- Полная функциональность Guardant Sign;
- Возможность загрузки собственных криптографических алгоритмов в ключ;
- Возможность выгрузки части кода защищаемого приложения в ключ для последующего исполнения непосредственно внутри ключа (с возможностью работы загруженного кода с защищенной областью памяти ключа).

Как упоминалось ранее, ключи Guardant Sign/Time Net могут быть использованы для лицензирования копий защищенного приложения в локальной сети.

Стоит отметить, что если функционал приложения может быть разделен на отдельно лицензируемые модули, то для каждого из модулей целесообразно использовать собственный секретный ключ, записывая, соответствующее число алгоритмов в маску ключа на этапе предпродажной подготовки. Более подробно этот вопрос будет рассмотрен в одном из следующих уроков.

Изучение уроков и документации

Теперь можно переходить к различного рода экспериментам и, в конечном итоге, реализации защиты. В процессе реализации будет необходимо неоднократно обращаться к документации. Она состоит из следующих частей:

- Руководство пользователя. Часть 1 (PDF) – описание утилит Мастер комплекта;
- Руководство пользователя. Часть 2 (PDF) – порядок работы с ключами Sign и Time;
- Руководство пользователя. Часть 3 (PDF) – порядок работы с ключами Code;
- Справка по Guardant API (CHM);
- Справка по Guardant API для платформы .NET (CHM);
- Руководство системного администратора (PDF) – порядок установки и настройки ПО сетевых ключей Guardant;
- Рекомендации по настройке ключей под ОС WinCE/Linux (PDF)

Исключительно полезным ресурсом при организации защиты на основе Guardant API являются примеры использования GrdAPI для различных языков программирования, расположенные в директории *“%Program Files% \Guardant\Guardant 5\%Public Code%\ Samples”*.

Подготовка защищенного приложения

Перед тем как приступить к реализации защиты продукта также необходимо:

1. Разработать ***стратегию лицензирования*** (будут ли все проданные лицензии равнозначны, или напротив, функционал каждой копии защищенного приложения будет разбит на отдельно лицензируемые модули);
2. ***Запрограммировать ключи*** в соответствии с выбранной стратегией лицензирования, сохранив информацию о прошивках в БД информации о клиентах;
3. ***Организовать*** службу технической поддержки и приступить к ***распространению*** защищенного приложения.

В процессе распространения защищенного приложения, при получении необходимой обратной связи от конечных пользователей, может возникнуть необходимость изменения/отзыва переданных лицензий или полное изменение всей стратегии лицензирования. Все это можно будет реализовать на основе возможностей, предоставляемых ***технологиями Guardant***.

Заключение

Уроки дают общее представление о процессе создания защиты и составлены в порядке возрастания сложности. В результате выполнения каждого из уроков будет получено защищенного приложение с использованием того или иного метода защиты.

За подробной информацией рекомендуется обращаться к документации. Она поставляется вместе с комплектом разработчика, а также доступна на сайте продукта.

Дополнительные источники информации

При возникновении вопросов, на которые вам не удалось найти ответа в этом пособии, рекомендуем обратиться к следующим дополнительным источникам информации:

WWW: http://www.guardant.ru

Web-сайт разработчика содержит большой объем справочной информации об электронных ключах Guardant.

Служба технической поддержки:

e-mail: hotline@guardant.ru

тел. +7(495)925-77-90