12. *Бородакий Ю.В.* Фундаментальные проблемы теории информационной безопасности автоматизированных систем // Труды VIII Международной конференции «Комплексная защита информации». 23–26 марта 2004. – Валдай.

А.В. Благодаренко
Россия, г. Таганрог, Технологический институт ЮФУ

ИНСТРУМЕНТАЛЬНОЕ СРЕДСТВО ДЛЯ ПРОВЕДЕНИЯ СЕРТИФИКАЦИОННЫХ ИСПЫТАНИЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ БЕЗ ИСХОДНЫХ КОДОВ

В настоящее время стремительного развития информационных технологий и глобальной компьютеризации человеческой жизни на передний план выходит зависимость человека от компьютерных программ. Их корректная работа и неподверженность удаленным атакам все больше и больше волнует пользователей, так как множество личной и ценной информации хранится на электронных носителях. Лабораторные сертификационные испытания призваны определить степень доверия пользователя к программному обеспечению. При проведении испытаний сертификационные лаборатории опираются на Руководящий документ (РД) Гостехкомиссии России по контролю над недекларированными возможностями [1]. Данный документ предписывает проводить анализ исследуемого ПО по исходным текстам. Однако существуют ситуации, когда исходные тексты для ПО недоступны. Например, если ПО распространяется свободно и автор находится за границей[2]. В такой ситуации сертификация может быть проведена и по исполняемому коду. В настоящее время, однако, для подобных испытаний нет соответствующей нормативной базы. Инструментальные средства для анализа ПО без исходных кодов есть, но они не удовлетворяют всем требованиям к проведению сертификационных испытаниях (если ввести в РД допущение на сертификацию ПО без исходных кодов). В табл. 1 рассматривается пригодность использования различных программных средств для проведения анализа ПО без исходных кодов.

Таблица 1

Пригодность существующих инструментальных средств для проведения сертификационных испытаний ПО без исходных кодов

Инструментальное средство	Недостатки
Аист - С	Ограниченность языком С/С++ Только статический анализ
EMU	Виртуальная среда Отсутствие статического и динамического анализа
SoftIce	Минимальный статистический анализ Отсутствие динамического анализа
IDA	Только статический анализ

Первые два средства из представленных в таблице («Аист-С» и «EMU») являются рекомендованными для проведения сертификационных испытаний ПО[2]. «Аист-С» представляет собой довольно мощный инструмент для анализа исходных кодов С/С++ с возможностью построения трасс, графов взаимодействия функциональных единиц и оценки избыточности. Выполнение тех же действий над ПО без исходных кодов требует сочетания разных инструментальных средств.

В комплекс должен входить, как минимум, дизассемблер и отладчик. Желательно также наличие каких-либо средств для анализа результата дизассемблирования. Стандартная связка для анализа исполняемого кода ПО «Softice + IDA». Numega Softice – продвинутый отладчик уровня ядра с возможностью установки точек останова и трассировки. IDA – интерактивный дизассемблер с развитым скриптовым языком, способный проводить базовый анализ дизассемблированного кода. Такая связка позволяет эффективно проводить автоматический статический анализ и ручной динамический. Для проведения автоматического анализа нужна более тесная связка дизассемблера и отладчика. На рис.1 представлена схема комплекса codeflow, первоначально разработанная для уменьшения объема кода, требующего ручного анализа [3,4].

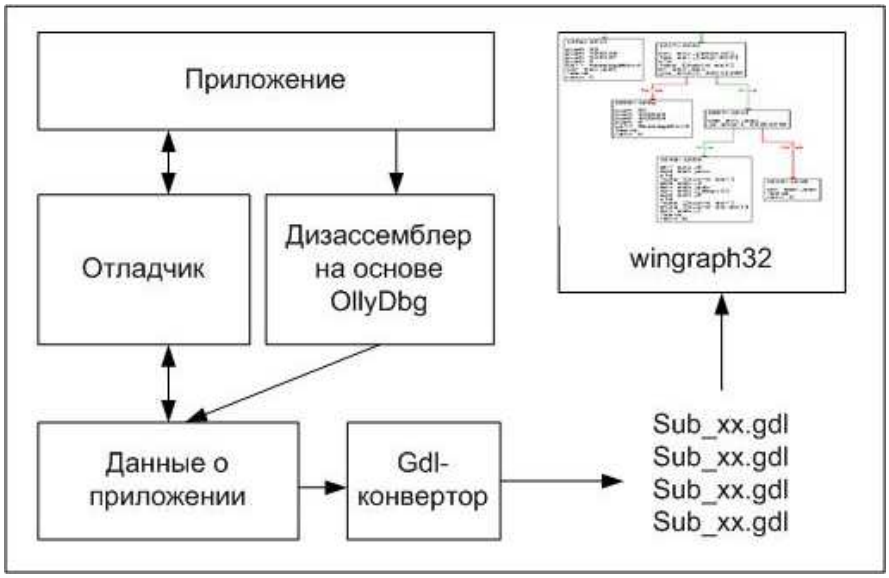

Рис. 1. Архитектура системы codeflow

При использовании разработанного средства тестируемое приложение запускается под управлением отладчика. Каждый подгружаемый в приложение модуль анализируется статически, после чего он подготавливается к динамическому анализу. В ходе динамического анализа собирается информация о приложении. В любой момент, по запросу пользователя, накопленные данные могут быть сохранены на диск в формате gdl(Graph Description Language,) а затем просмотрен во внешней программе, поддерживающей данный формат, например wingraph32. Данные представляются в виде деревьев возможных путей исполнения программы с дизассемблированными листингами (используется код дизассемблера Ollydbg). Деревья содержат данные, собранные в ходе динамического анализа, такие как части кода, анализ которых невозможно выполнить статически, а также частота исполнения кода.

Как видно из рис.1 система состоит из набора модулей. Такая архитектура более гибка: состав модулей может меняться, отдельные модули могут быть вновь переписаны. Монолитные же решения, такие как “Аист-С”, сложно поддерживать в актуальном состоянии. На рис. 2 представлена архитектура для построения маршрутов исполнения программы, основанная на вышеописанной.

Рис. 2. Архитектура системы построения маршрутов исполнения

Под маршрутом понимается последовательность команд, выполняющихся при отработке какого-либо действия, входящего в функциональность программы. Например, если ПО обрабатывает файл определенного формата, то можно говорить о маршрутах, выполняемых при его разборке и выполнении.

Для анализатора исходных текстов «АИСТ-С» маршрут – это последовательность команд языка С или С++, для системы, работающей с исполняемым кодом – последовательность машинных команд.

Моделированием воздействий на исследуемое ПО занимаются генератор контента и модуль воздействия. Далее, опираясь на данные от отладчика и дизассемблера, отслеживается реакция на входное воздействие.

Подобная система может быть использована для автоматизированного поиска уязвимостей и проведения сертификационных испытаний.

Работа поддержана грантом РФФИ 07-07-00138-а, 06-07-89010.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1.Руководящий документ. Защита от несанкционированного доступа к информации. Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей. – М.: Гостехкомиссия России, 1998.

2.*Марков, А.С.* Выявление уязвимостей в программном коде /А.С. Марков, С.В. Миронов, В.Л. Цирлов // [Интернет].- режим доступа
http://www.osp.ru/os/2005/12/380655/_p2.html, свободный

3.*Благодаренко, А.В.* Статистический и динамический анализ программного обеспечения без исходных текстов // Проблемы информационной безопасности в системе высшей школы: труды XIV всероссийской научной конференции.–М.:МИФИ, 2007.-С. 33-34.

4.*Бойко А.Н.* Выделение редко исполняемого кода. Сб.трудов VII Всероссийской научной конференции студентов и аспирантов «Техническая кибернетика, радиоэлектроника и системы управления», Таганрог 2006. – С. 334.

М.Е. Путивцев, А.А. Баранник
Россия, г. Таганрог, Технологический Институт ЮФУ

МОДЕЛЬ ПРОВЕДЕНИЯ СЕРТИФИКАЦИИ ПО СТАНДАРТУ ISO\IEC 17799 С ИСПОЛЬЗОВАНИЕМ ПРОЦЕССНОГО ПОДХОДА

Обеспечение информационной безопасности (ИБ) компьютерных систем различного назначения продолжает оставаться чрезвычайно острой проблемой. Можно констатировать тот факт, что несмотря на усилия многочисленных организаций, занимающихся решением этой проблемы, общая тенденция остается негативной. Основных причин этому две:

– возрастающая роль информационных технологий в поддержке бизнес-процессов, как следствие возрастающих требования к ИБ автоматизированных систем. Цена ошибок и сбоев информационных систем возрастает;

– возрастающая сложность информационных процессов. Это предъявляет повышенные требования к квалификации персонала, ответственного за обеспечение ИБ. Выбор адекватных решений, обеспечивающих приемлемый уровень ИБ при допустимом уровне затрат, становится все более сложной задачей.

Для решения этих задач создаются организации аудиторов в области ИБ, ставящие своей целью проведение экспертизы соответствия системы ИБ некоторым требованиям, оценки системы управления ИБ, повышения квалификации специалистов в области ИБ [1]. Система аттестации обычно появляется одновременно с принятием стандартов ИБ.

В последнее время в разных странах появилось новое поколение стандартов в области ИБ, посвященных практическим аспектам организации и управления ИБ [2].

В данной статье на этапе анализа методологий оценки уровня информационной безопасности, ввиду доступности и достаточно широкой распространенности был отдан приоритет международному стандарту ISO\IEC 17799 [3].

Технология проведения аудита на соответствие подобным стандартам существенно отличается от технологий, применяемых для предыдущих поколений стандартов, к которым, в частности, относятся Руководящие документы (РД) Гостехкомиссии России 1992–1993 гг. Основное отличие заключается в гораздо большей степени формализации некоторых этапов, использовании поддающихся проверке показателей и критериев, то есть в большей детализации.

Однако при рассмотрении стандарта ISO\IEC 17799 необходимо отметить, что при проведении сертификации на соответствие степень формализации оценочных работ явно недостаточна. Для решения данной проблемы было решено про-