Бородин Андрей Викторович
канд. экон. наук, доцент, заведующий кафедрой
Старовойтов Алексей Андреевич
студент
ФГБОУ ВО «Поволжский государственный
технологический университет»
г. Йошкар-Ола, Республика Марий Эл

DOI 10.21661/r-464620

О ЗАДАЧЕ КОНТРОЛЯ ЦЕЛОСТНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ УДАЛЕННОГО НЕДОВЕРЕННОГО ХОСТА

Аннотация*: в статье обоснована неразрешимость задачи дистанционного контроля целостности программного обеспечения недоверенного хоста. Показана возможность сведения атаки на целостность к атаке на результат обфускации кода.*

Ключевые слова*: доверенная аппаратная компонента, доверенный хост, изолированная программная среда, криптографическая хеш-функция, неразрешимость, обфускация, удаленный хост, целостность.*

В настоящее время распределенные приложения становятся обычным делом. Это могут быть системы GRID-вычислений [11], различные P2P-среды [9]. Во многих случаях, если не во всех, для таких систем актуальна задача контроля целостности программного обеспечения (ПО) хоста, относительно которого справедливо предположение о невозможности обеспечить на нем изолированную программную среду (ИПС) [4]. Данная работа посвящена формальному анализу этой задачи.

Итак, *доверенным хостом* будем называть хост, на котором функционирует ИПС, обеспечивающая доверенную загрузку компонент ПО, то есть такой режим работы системы, когда загрузку очередного компонента ПО и передачу ему управления обязательно предваряет контроль целостности этого компонента со

стороны уже функционирующего. При этом предполагается, что в системе присутствует доверенная аппаратная компонента, которая проверит те компоненты системы, которые участвуют в начальной загрузке компьютера [6]. Несоблюдение любого из перечисленных условий функционирования ИПС делает хост *недоверенным*. Далее будем предполагать, что в качестве средства *контроля целостности* данных используется механизм сравнения значения криптографической хеш-функции (КХФ) [10] от этих данных с соответствующим эталоном. И, наконец, под *реверс-инжинирингом* будем понимать исследование некоторого готового устройства или программы, а также документации на него с целью понять принцип его работы, например, чтобы обнаружить недокументированные возможности (в том числе программные закладки), сделать изменение или воспроизвести устройство, программу или иной объект с аналогичными функциями, но без прямого копирования [8]. Используя введенные понятия, можно доказать следующее утверждение.

Утверждение 1. Контроль целостности ПО удаленного недоверенного хоста в условиях неограниченного ресурса злоумышленника на реверс-инжиниринг не возможен.

Доказательство этого утверждения будет носить конструктивный характер.

Пусть функция $f: X \to Y$ описывает прикладную роль удаленного хоста. Здесь X – множество данных, на которых может быть реализована функциональность данного хоста, Y – множество возможных результатов реализации хостом своей функциональности. Эти множества можно интерпретировать, например, следующим образом: X – множество возможных запросов к хосту, а Y – множество возможных ответов на эти запросы.

Также предположим, что хост может вычислять значения КХФ различных компонент своего ПО. Эту возможность опишем функцией $h: C \times \mathfrak{A} \to H$, где C – множество идентификаторов компонент ПО хоста, $\mathfrak{A}$ – множество возможных вариантов ПО хоста, H – множество возможных значений КХФ.

Пользуясь введенными обозначениями, модель удаленного хоста можно представить в виде функции ответов хоста на все возможные запросы:

$$A(x)=\left\{\begin{array}{c} f(x) \\ h(x,A) \end{array}\middle|\begin{array}{l} x\in X \\ x\in C \end{array}\right\}.$$

Здесь мы предполагаем, что $X\cap C=\varnothing$.

Заметим, что неизменность множества

$$\left\{\left(x,h(x,A)\right):\ x\in C\right\}$$

может гарантировать целостность реализации A и, следовательно f.

В условиях недоверенности удаленного хоста злоумышленник, пользуясь неограниченным ресурсом реверс-инжиниринга реализации функции A, может модифицировать ПО таким образом, что вместо A будет реализована другая функция:

$$\tilde{A}(x)=\left\{\begin{array}{c} \tilde{f}(x) \\ h(x,A) \end{array}\middle|\begin{array}{l} x\in X \\ x\in C \end{array}\right\},$$

где $\tilde{f}\neq f$. Однако, при этом

$$\left\{\left(x,\tilde{A}(x)\right):\ x\in C\right\}=\left\{\left(x,h(x,A)\right):\ x\in C\right\}.$$

Последнее соотношение фактически *доказывает* Утверждение 1.

Заметим, что осуществление атаки, возможность которой декларирует утверждение 1, требует от злоумышленника, во-первых, классификации внешних запросов к хосту в части принадлежности множеству X или множеству C, во-вторых, реализации новой функциональности $\tilde{f}$, в-третьих, эмуляции исполнения A, и, наконец, диспетчеризации сообщений в смысле реализации $\tilde{A}$. Это даже не требует полного реверс-инжиниринга A.

В то же время соблазн обфускации кода, реализующего КХФ, и создания на этой основе механизма обнаружения атак на целостность ПО удаленного недоверенного хоста велик. Термин «обфускация» мы будем понимать здесь в смысле определения 1 из работы [1], см., также, [7].

Сформулируем новое утверждение.

Утверждение 2. Задача атаки на целостность ПО удаленного недоверенного хоста может быть эквивалентна задаче реверс-инжиниринга некоторой части этого ПО.

Доказательство этого утверждения проведем также в русле конструктивизма.

Пусть в начальный момент $y = y_0$, где y_0 – некоторая константа. Модель удаленного хоста будем представлять в виде функции ответов хоста на все возможные запросы:

$$A^*(x) = \left\{ \begin{matrix} f(x) \\ h^*(y, x, A) \end{matrix} \middle| \begin{matrix} x \in X \\ x \in C \end{matrix} \right\}, \; X \cap C = \varnothing.$$

При этом реализация f такова, что она имеет неявный побочный эффект присвоения $y := f(x)$ с точки зрения реализации функции h^*. В то же время, функция $h^* : Y \times C \times \mathfrak{A} \to H$ является необратимой [3, с. 71] в том смысле, что вычисление $h^*(y, x, A)$ при известных y, x и A не является трудоемким, а поиск решений уравнения

$$h^*(y', x, A') = h'$$

относительно неизвестных y' и A' при известных x и h' вычислительно крайне трудоемок.

В описанной ситуации атака злоумышленника на целостность ПО может быть представлена реализацией функции $\tilde{A}^*$ вместо A^*, такой, что

$$\tilde{A}^*(x) = \left\{ \begin{matrix} \tilde{f}(x) \\ h^*(y, x, A) \end{matrix} \middle| \begin{matrix} x \in X \\ x \in C \end{matrix} \right\},$$

где также как и раньше $\tilde{f} \neq f$, а неявный побочный эффект реализации функции $\tilde{f}$ с точки зрения реализации функции h^* должен заключаться в присвоении $y := \tilde{f}(x)$.

Таким образом, в описанной постановке реализация атаки на целостность ПО хоста потребует новой реализации вычисления значений функции h^*, учитывающей новый побочный эффект вычисления $\tilde{f}$, что необходимо влечет потребность в реверс-инжиниринге исходной реализации h^*. Утверждение 2 *доказано*.

Заметим, что эффективным подходом к реализации требуемого побочного эффекта может стать технология рандомизации карты памяти программы [5], а сама обфускация кода, реализующего h^*, может быть основана на эквивалентных преобразованиях программ, предложенных в работах [1; 2; 7].

Подводя итог, отметим, что задача контроля целостности ПО удаленного недоверенного хоста не имеет строго математически обоснованного решения, в данной статье было приведено доказательство этого утверждения. В то же время показано, что достижима ситуация, когда указанная атака с необходимостью влечет задачу реверс-инжиниринга определенного фрагмента кода. Это позволяет свести задачу атаки на целостность ПО хоста к задаче атаки на результат применения некоторой технологии обфускации. Последний факт может быть эффективно использован для реализации механизмов обнаружения атак на целостность в вероятностной постановке.

Список литературы

1. Бородин А.В. Линейные конгруэнтные последовательности максимального периода в задачах обфускации программ / А.В. Бородин // Кибернетика и программирование. – 2016. – №6. – С. 1–19. – DOI: 10.7256/2306–4196.2016.6.18499.

2. Бородин А.В. Обфускация пула констант как задача построения минимальной системы целочисленных линейных комбинаций / А.В. Бородин, Е.Д. Долгушев // Образование, наука, бизнес: развитие и перспективы: Материалы III международной научно-практической конференции (6 мая 2016 г.). – Саратов: Изд-во ЦПМ «Академия Бизнеса», 2016. – С. 8–13.

3. Бородин А.В. Феномен компьютерных вирусов: элементы теории и экономика существования / А.В. Бородин. – Йошкар-Ола: Марийский государственный технический университет, 2004. – 144 с.

4. Девянин П.Н. Модели безопасности компьютерных систем. Управление доступом и информационными потоками / П.Н. Девянин. – М.: Техносфера, 2013. – 338 с.

5. Козлова К.А. Пул переменных программы как объект-хранилище с рандомизацией карты памяти / К.А. Козлова, А.В. Бородин // Инженерные кадры – будущее инновационной экономики России: Материалы II Всероссийской студенческой конференции (Йошкар-Ола, 21–25 ноября 2016 г.). – Йошкар-Ола: Поволжский государственный технологический университет, 2016. – №4. – С. 49–51.

6. Конявский В.А. Управление защитой информации на базе СЗИ НСД «Аккорд» / В.А. Конявский. – М.: Радио и связь, 1999. – 325 с.

7. Львович И.Я. Перспективные тренды развития науки: техника и технологии: Т. 1 / И.Я. Львович, В.А. Некрасов, А.П. Преображенский [и др.]. – Одесса: Куприенко С.В., 2016 – 197 с.

8. Обратная разработка // Википедия. Свободная энциклопедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%80%D0%B0%D1%82%D0%BD%D0%B0%D1%8F_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0.

9. P2P – Следующий этап развития информационных систем // Хабрахабр [Электронный ресурс]. – Режим доступа: https://habrahabr.ru/post/239225/.

10. Katz J. Introduction to Modern Cryptography. Second Edition / J. Katz, Y. Lindell. – CRC Press, 2015. – 603 p.

11. Maozhen L. The Grid: Core Technologies / L. Maozhen, M. Baker. – John Wiley & Sons, Ltd, 2005. – 452 p.