РОССИЙСКАЯ ФЕДЕРАЦИЯ

ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2510075

СПОСОБ ОБНАРУЖЕНИЯ ВРЕДОНОСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ЯДРЕ ОПЕРАЦИОННОЙ СИСТЕМЫ

Патентообладатель(ли): ***Открытое акционерное общество "Информационные технологии и коммуникационные системы" (RU)***

Автор(ы): ***см. на обороте***

Заявка № **2012113963**

Приоритет изобретения **11 апреля 2012 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации ***20 марта 2014 г.***

Срок действия патента истекает **11 апреля 2032 г.**

Руководитель Федеральной службы по интеллектуальной собственности

Б.П. Симонов

РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) 2 510 075 (13) **C2**

(51) МПК
G06F 21/56 (2013.01)
G06F 12/14 (2006.01)
G06F 11/00 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ФОРМУЛА ИЗОБРЕТЕНИЯ К ПАТЕНТУ РОССИЙСКОЙ ФЕДЕРАЦИИ

(21)(22) Заявка: 2012113963/08, 11.04.2012

(24) Дата начала отсчета срока действия патента:
11.04.2012

Приоритет(ы):
(22) Дата подачи заявки: 11.04.2012

(43) Дата публикации заявки: 20.10.2013 Бюл. № 29

(45) Опубликовано: 20.03.2014 Бюл. № 8

(56) Список документов, цитированных в отчете о поиске: RU 107619 U1, 20.08.2011. RU 107620 U1, 20.08.2011. US 7571482 B2, 04.08.2009. US 7647636 B2, 12.01.2010. US 8104089 B1, 24.01.2012. US 7673137 B2, 02.03.2010.

Адрес для переписки:
127287, Москва, Старый Петровско-Разумовский пр-д, 1/23, стр.1, Открытое акционерное общество "Информационные технологии и коммуникационные системы"

(72) Автор(ы):
Тумоян Евгений Петрович (RU),
Ольшанов Константин Дмитриевич (RU),
Черемнцев Сергей Николаевич (RU)

(73) Патентообладатель(и):
Открытое акционерное общество "Информационные технологии и коммуникационные системы" (RU)

RU 2510075 C2

(54) СПОСОБ ОБНАРУЖЕНИЯ ВРЕДОНОСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ЯДРЕ ОПЕРАЦИОННОЙ СИСТЕМЫ

(57) Формула изобретения

Способ обнаружения вредоносного программного обеспечения в ядре операционной системы, установленной на компьютере, заключающийся в том, что

формируют точку прерывания при выполнении системного вызова пользовательского приложения на возникновение передачи управления по адресу в ядре загруженной ОС,

проводят проверку структуры данных загруженной операционной системы, выполняя следующие действия:

определяют адрес команды в оперативной памяти компьютера, которой будет передано управление в ходе системного вызова;

проверяют принадлежность адресов команд, выполняемых в ходе системного вызова, к нормальному диапазону адресов ядра и модулей ядра операционной системы в оперативной памяти;

судят о наличии вредоносного программного обеспечения при отсутствии принадлежности адреса команды к нормальному диапазону адресов.

РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 510 075** (13) **C2**

(51) МПК
G06F 21/56 (2013.01)
G06F 12/14 (2006.01)
G06F 11/00 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: **2012113963/08, 11.04.2012**

(24) Дата начала отсчета срока действия патента:
11.04.2012

Приоритет(ы):
(22) Дата подачи заявки: **11.04.2012**

(43) Дата публикации заявки: **20.10.2013** Бюл. № 29

(45) Опубликовано: **20.03.2014** Бюл. № 8

(56) Список документов, цитированных в отчете о поиске: **RU 107619 U1, 20.08.2011. RU 107620 U1, 20.08.2011. US 7571482 B2, 04.08.2009. US 7647636 B2, 12.01.2010. US 8104089 B1, 24.01.2012. US 7673137 B2, 02.03.2010.**

Адрес для переписки:
127287, Москва, Старый Петровско-Разумовский пр-д, 1/23, стр.1, Открытое акционерное общество "Информационные технологии и коммуникационные системы"

(72) Автор(ы):
Тумоян Евгений Петрович (RU),
Ольшанов Константин Дмитриевич (RU),
Черемнцев Сергей Николаевич (RU)

(73) Патентообладатель(и):
Открытое акционерное общество "Информационные технологии и коммуникационные системы" (RU)

RU 2510075 C2

(54) СПОСОБ ОБНАРУЖЕНИЯ ВРЕДОНОСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ЯДРЕ ОПЕРАЦИОННОЙ СИСТЕМЫ

(57) Реферат:

Изобретение относится к вычислительной технике и к обеспечению информационной безопасности автоматизированных и информационно-вычислительных систем, в частности к средствам обнаружения вредоносного программного обеспечения (ПО). Техническим результатом является повышение эффективности обнаружения вредоносного ПО за счет обеспечения возможности обнаружения нелегальных перехватов и изменения кода в ядре и загружаемых модулях ядра ОС. Способ реализуется на компьютере с установленной на нем операционной системой (ОС) и заключается в том, что формируют точку прерывания при выполнении системного вызова пользовательского приложения на возникновение передачи управления по адресу в ядре загруженной ОС, проводят проверку структуры данных загруженной ОС, выполняя следующие действия: определяют адрес команды в оперативной памяти компьютера, которой будет передано управление в ходе системного вызова; проверяют принадлежность адресов команд, выполняемых в ходе системного вызова, к нормальному диапазону адресов ядра и модулей ядра ОС в оперативной памяти; судят о наличии вредоносного ПО при отсутствии принадлежности адреса команды к нормальному диапазону адресов.

RU 2510075 C2

Область техники, к которой относится изобретение

Изобретение относится к вычислительной технике и к обеспечению информационной безопасности автоматизированных и информационно-вычислительных систем, в частности к средствам обнаружения вредоносного программного обеспечения.

Уровень техники

В настоящее время разработаны и реализованы в прикладном программном обеспечении (ПО) следующие основные методы обнаружения вредоносного ПО:

1) обнаружение фактов скрытия программ, файлов, процессов, модулей ядра путем проверки наличия данных объектов на разных уровнях операционной системы (ОС);

2) обнаружение отличительных признаков ранее зарегистрированного вредоносного ПО, в частности последовательностей байтов исполняемого кода, строк и констант, характерных для вредоносных программ;

3) контроль целостности исполняемого кода ядра в оперативной памяти.

Так, известен прикладной программный пакет Rootkit Profiler LX для обнаружения вредоносного ПО [1].

Средство Rootkit Profiler LX выполняет проверку:

1) адреса таблицы системных вызовов, таблицы прерываний, обработчика системного вызова, обработчика прерывания,

2) кода системного вызова, кода обработчика прерывания,

3) указателей в структурах виртуальной файловой системы (Virtual File System, VFS).

Таким образом, известное средство не проводит проверку целостности всего кода ядра, а также динамическую проверку исполнения кода ядра, вследствие чего не обнаруживает перехватов в структурах данных ОС, а также перехватов и модификации кода в ядре (за исключением структур виртуальной файловой системы), что является недостатком.

Известен также способ обнаружения вредоносного ПО [2], реализуемый на компьютере с ОС и, согласно одному из вариантов реализации, заключающийся в том, что:

- формируют точку прерывания при возникновении системного вызова из пользовательского приложения на изменение структуры данных загруженной ОС;

- проводят проверку структуры данных загруженной ОС, выполняя следующие действия:

- определяют адрес команды в оперативной памяти компьютера, которая произвела изменения в структуре данных;

- проверяют принадлежность адреса команды к нормальному диапазону адресов ОС в оперативной памяти;

- судят о наличии ВП при отсутствии принадлежности адреса команды к нормальному диапазону адресов.

Под структурой данных здесь понимаются формируемые ОС таблицы исполняемых процессов (в том числе пользовательских приложений), ссылки на системный реестр и файлы и пр.

Перед реализацией способа обеспечивают доступ на чтение и запись к областям оперативной памяти с загруженным ядром ОС и модулями ядра и загружают ОС на компьютер. Способ может быть реализован на компьютерах с ОС общего назначения типа Unix, Linux, Microsoft Windows и др.

Непосредственная реализация способа обеспечивается с помощью предварительно создаваемого прикладного ПО:

- средства отладчика ядра (kernel debugger facilities), выполненного с возможностью получать данные из структур данных, сформированных ОС, путем установки точки прерывания;

- модуля проверки целостности (integrity checker), выполненного с возможностью определения, содержат ли данные, полученные средствами отладчика ядра, когда точка прерывания была установлена, несогласованности, характерные для вредоносного ПО;

- модуль обнаружения (detection module), который координирует полученные данные, сформированные ОС, и обеспечивает данные для модуля проверки целостности.

Перечисленные программные средства не являются уникальными и могут быть созданы профильным специалистом (программистом) на основе знания выполняемых этими средствами функций.

Известный способ принят за прототип для предлагаемого технического решения.

Недостатком известного способа является невысокая вероятность обнаружения вредоносного ПО, поскольку обеспечивается отслеживание действий только при изменении в структуре данных ОС. Соответственно, вредоносное ПО непосредственно в ядре ОС и модулях ядра при использовании известного способа не будет обнаружено.

Раскрытие изобретения

Предлагаемый способ включает динамическую проверку исполнения кода ядра ОС для обнаружения нелегальных перехватов и изменения кода в ядре и загружаемых модулях ядра (драйверах).

Техническим результатом является повышение вероятности обнаружения вредоносного ПО за счет обеспечения возможности обнаружения нелегальных перехватов и изменения кода в ядре и загружаемых модулях ядра ОС.

Дополнительным техническим результатом является также обеспечение проверки целостности исполняемого кода ядра.

Для этого предлагается способ, заключающийся в том, что

- формируют точку прерывания при выполнении системного вызова пользовательского приложения на возникновение передачи управления по адресу в ядре загруженной ОС;

- проводят проверку структуры данных загруженной ОС, выполняя следующие действия:

- определяют адрес команды в оперативной памяти компьютера, которой будет передано управление в ходе системного вызова;

- проверяют принадлежность адресов команд, выполняемых в ходе системного вызова, к нормальному диапазону адресов ядра и модулей ядра ОС в оперативной памяти;

- судят о наличии ВП при отсутствии принадлежности адреса команды к нормальному диапазону адресов.

Таким образом, в отличие от известного способа, в котором проверка проводится только в случае изменения в структуре данных загруженной ОС, в предлагаемом способе проводится проверка всей совокупности адресов команд, осуществляемых в ходе системного вызова.

Это позволяет контролировать все команды и переходы по адресам, в том числе генерируемые вредоносным ПО в составе ядра и модулей ОС, а не только вредоносным ПО в пользовательских приложениях.

Дополнительно также выявляется целостность исполняемого кода ядра.

Осуществление изобретения

Необходимым условием реализации предложенного способа является обеспечение доступа на чтение и запись к областям оперативной памяти с загруженным ядром ОС и модулями ядра. Это условие наиболее просто реализуется в ОС Linux.

Процесс осуществления предлагаемого способа далее описывается для компьютера, имеющего:

1) ОС Linux Ubuntu 9.10 с ядром серии 2.6.31,

2) процессор Pentium серии P6 и более поздних производства компании Intel (США).

Для автоматизированного выполнения предлагаемого способа необходимо, как и в прототипе, предварительно сформировать программу средства отладчика ядра, позволяющую

1) устанавливать точки прерывания на команды передачи управления;

2) определять адреса команды в оперативной памяти, на которую будет передано управление.

Создание такой программы может осуществить специалист в области ОС (программист).

Перед непосредственным осуществлением предлагаемого способа целесообразно обеспечить наличие доверительного образа ядра ОС для последующего сравнения.

Для этого сначала формируют образ ядра ОС, например, на жестком диске компьютера, а затем распаковывают образ ядра. Обычно код ядра упакован с помощью какой-либо известной программы-архиватора, при упаковке/распаковке могут использоваться алгоритмы Gzip, Bzip и др.

После распаковки код ядра представляет собой файл формата .elf. В полученном файле производится определение сегментов кода.

В сегментах кода выполняется исправление изменяемых адресов кода ядра аналогично тому, как это выполняет стандартный для данной ОС загрузчик ядра:

- изменение альтернативных инструкций - инструкций, указанных в секции .altinstructions ядра,
- изменение инструкций с префиксом LOCK,
- изменение инструкций, связанных с паравиртуализацией.

В результате получают доверительный (эталонный) образ ядра ОС и нормальный диапазон адресов размещения кода ядра в оперативной памяти.

Получение доверительного образа ядра ОС производится с использованием стандартных средств работы с файлами ОС.

Для получения нормального диапазона адресов ядра ОС и модулей ядра необходимо также предварительно сформировать вспомогательную программу, позволяющую:

1) сформировать доверительный образ ядра в оперативной памяти;

2) определять адрес начала и конца каждого сегмента ядра в оперативной памяти.

Целесообразность создания такой вспомогательной программы определяется тем, что стандартные средства ОС могут быть модифицированы ВП для скрытия своего наличия в компьютере.

Создание такой вспомогательной программы также может осуществить специалист в области ОС (программист).

После выполнения перечисленных выше подготовительных действий, в первую очередь осуществляют проверку целостности кода ядра загруженной ОС в оперативной памяти компьютера путем последовательного сравнения с

доверительным кодом ядра.

Если обнаружено изменение адреса, проводится проверка, принадлежит ли адрес доверительному модулю в составе ядра. Если нет - считают, что обнаружено вредоносное ПО.

Если обнаружено изменение команд - то также считают, что обнаружено вредоносное ПО.

При обнаружении вредоносного ПО для его последующей нейтрализации могут быть применены какие-либо известные методы, например последовательно выполняемые действия: завершение процесса, связанного с вредоносным ПО, удаление вредоносного ПО и удаление файла, который содержит код программы, которая осуществляет действия вредоносного ПО [2].

После успешной проверки целостности кода ядра выполняют динамическую проверку исполнения кода ядра.

Для этого формируют список адресов, соответствующих участкам кода ядра и модулей ядра.

Затем с помощью средства отладчика ядра блокируют переключение процессов.

Процессор переводят в отладочный режим и устанавливают флаг трассировки TF=1 для всех потоков ядра.

Отключают маскируемые прерывания инструкцией CLI.

Устанавливают перехватчики точек входа в ядро (SYSENTER, int80h).

Включают маскируемые прерывания инструкцией STI.

После этого разблокируют переключение процессов.

При выполнении отладочного прерывания по факту совершения перехода выполняют проверку адреса перехода.

Если адрес не принадлежит коду ядра или коду модуля ядра, то считают модуль, который содержит данный адрес, вредоносным.

Затем устанавливают в процессоре флаг BTF=1 и выходят из обработчика отладочного прерывания.

Возможные действия при обнаружении вредоносного модуля ядра для его последующей нейтрализации могут быть аналогичны описанным выше.

Источники информации

1. Klein T. - Rootkit Profiler LX. Overview and Documentation, 2007 [Электронный ресурс]. - Режим доступа:
http://www.trapkit.de/research/rkprofiler/rkplx/RKProfilerLX_v0.12_20070422.pdf, дата обращения 09.03.2012, свободный.

2. Патент США №7571482, Automated rootkit detector, приоритет от 28.06.2005 г.

Формула изобретения

Способ обнаружения вредоносного программного обеспечения в ядре операционной системы, установленной на компьютере, заключающийся в том, что

формируют точку прерывания при выполнении системного вызова пользовательского приложения на возникновение передачи управления по адресу в ядре загруженной ОС,

проводят проверку структуры данных загруженной операционной системы, выполняя следующие действия:

определяют адрес команды в оперативной памяти компьютера, которой будет передано управление в ходе системного вызова;

проверяют принадлежность адресов команд, выполняемых в ходе системного

вызова, к нормальному диапазону адресов ядра и модулей ядра операционной системы в оперативной памяти;

судят о наличии вредоносного программного обеспечения при отсутствии принадлежности адреса команды к нормальному диапазону адресов.