



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

G06F 21/50 (2017.08); G06F 21/56 (2017.08)

(21)(22) Заявка: 2016115859, 25.09.2014

(24) Дата начала отсчета срока действия патента:
25.09.2014

Дата регистрации:
02.03.2018

Приоритет(ы):

(30) Конвенционный приоритет:
27.09.2013 US 14/040,430

(43) Дата публикации заявки: 30.10.2017 Бюл. № 31

(45) Опубликовано: 02.03.2018 Бюл. № 7

(85) Дата начала рассмотрения заявки РСТ на
национальной фазе: 27.04.2016

(86) Заявка РСТ:
RO 2014/000028 (25.09.2014)

(87) Публикация заявки РСТ:
WO 2015/171007 (12.11.2015)

Адрес для переписки:
191002, Санкт-Петербург, а/я 5, ООО "Ляпунов
и партнёры"

(72) Автор(ы):

МИРЧЕСКУ Даныел-Александру (RO)

(73) Патентообладатель(и):

БИТДЕФЕНДЕР АйПиАр
МЕНЕДЖМЕНТ ЛТД (CY)

(56) Список документов, цитированных в отчете
о поиске: US 8225406 B1, 17.07.2012. US 2011/
0185429 A1, 28.07.2011. US 8495705 B1,
23.07.2013. US 2004/0083381 A1, 29.04.2004. US
2008/0235801 A1, 25.09.2008. US 7392544 B1,
24.06.2008. US 2013/0254880 A1, 26.09.2013. US
2012/0192275 A1, 26.07.2012. US 8001606 B1,
16.08.2011. RU 2011138462 A, 10.04.2013.

(54) СИСТЕМА И СПОСОБ ДЛЯ ПРИМЕНЕНИЯ ИНДИКАТОРА РЕПУТАЦИИ ДЛЯ ОБЛЕГЧЕНИЯ
СКАНИРОВАНИЯ НА НАЛИЧИЕ ВРЕДОНОСНЫХ ПРОГРАММ

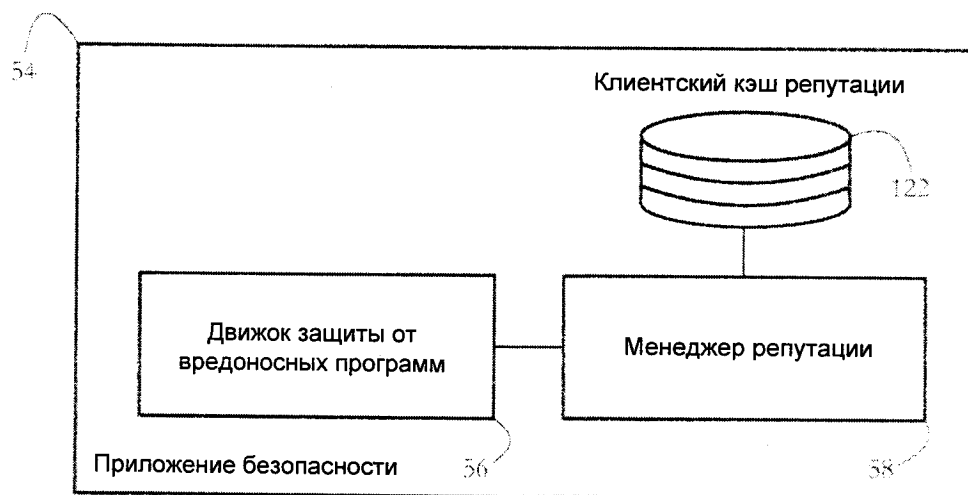
(57) Реферат:

Изобретение относится к защите компьютерной системы от вредоносных программ. Технический результат заключается в обеспечении быстрого, надежного и настраиваемого решения по защите от вредоносных программ. Клиентская система содержит аппаратный процессор, сконфигурированный для выполнения движка защиты от вредоносных программ, сконфигурированного для мониторинга целевого процесса на наличие вредоносной активности, причем целевой процесс содержит экземпляры

главного исполняемого модуля и общей библиотеки; приема от сервера первого индикатора репутации модуля главного исполняемого модуля и второго индикатора репутации модуля общей библиотеки, причем первый и второй индикаторы репутации модуля содержат соответственно первый и второй наборы правил мониторинга; определения, является ли целевой процесс вероятно вредоносным, в соответствии с первым и вторым индикаторами репутации модуля, и если целевой процесс вероятно не является вредоносным,

комбинирования первого и второго набора правил мониторинга в комбинированный набор правил мониторинга; и конфигурирования движка защиты от вредоносных программ для

мониторинга целевого процесса в соответствии с комбинированным набором правил мониторинга. 3 н. и 26 з.п. ф-лы, 21 ил., 2 табл.



Фиг. 6



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(52) CPC

G06F 21/50 (2017.08); *G06F 21/56* (2017.08)(21)(22) Application: **2016115859, 25.09.2014**(24) Effective date for property rights:
25.09.2014Registration date:
02.03.2018

Priority:

(30) Convention priority:
27.09.2013 US 14/040,430(43) Application published: **30.10.2017 Bull. № 31**(45) Date of publication: **02.03.2018 Bull. № 7**(85) Commencement of national phase: **27.04.2016**(86) PCT application:
RO 2014/000028 (25.09.2014)(87) PCT publication:
WO 2015/171007 (12.11.2015)Mail address:
**191002, Sankt-Peterburg, a/ya 5, OOO "Lyapunov i
partnery"**

(72) Inventor(s):

MIRCHESKU Danel-Aleksandru (RO)

(73) Proprietor(s):

**BITDEFENDER AjPiAr MENEDZHMENT
LTD (CY)**(54) **SYSTEMS AND METHODS FOR USING A REPUTATION INDICATOR TO FACILITATE MALWARE SCANNING**

(57) Abstract:

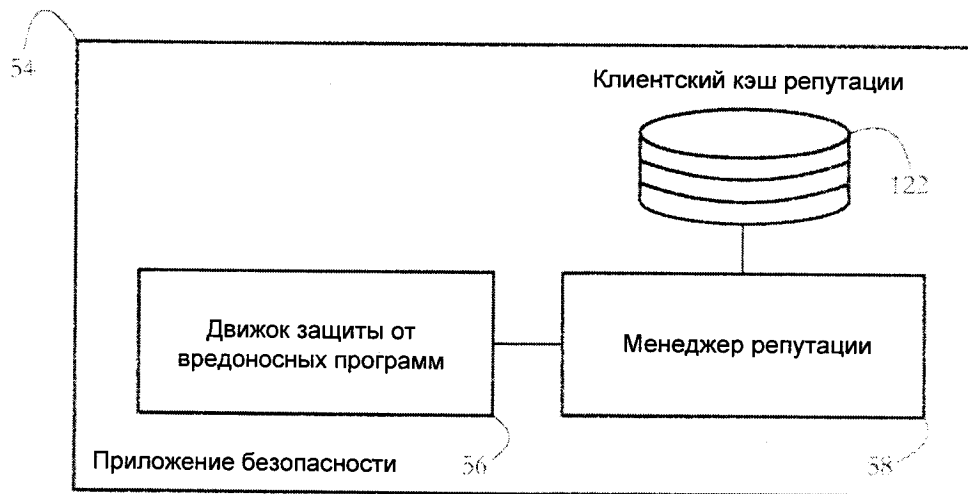
FIELD: data processing.

SUBSTANCE: invention relates to the protection of a computer system from malicious programs. Client system contains a hardware processor configured to run the malware protection engine, configured to monitor the target process for malicious activity, the target process contains instances of the main executable module and the shared library; receiving from the server the first indicator of the reputation of the module of the main executable module and the second indicator of the reputation of the common library module, the first and second reputation indicators of the module contain respectively the first and second sets of monitoring

rules; determining whether the target process is probably malicious, in accordance with the first and second indicators of the reputation of the module, and if the target process is probably not harmful, combining the first and second set of monitoring rules into a combined set of monitoring rules; and configuring the malware protection engine to monitor the target process in accordance with a combined set of monitoring rules.

EFFECT: technical result is to provide a fast, reliable and customizable solution to protect against malicious programs.

29 cl, 21 dwg, 2 tbl



Фиг. 6

Уровень техники

Данное изобретение относится к устройствам и способам защиты компьютерных систем от вредоносных программ.

Вредоносное программное обеспечение, известное также как вредоносные программы, наносит ущерб большому количеству вычислительных систем во всем мире. В ряде своих форм, таких, например, как компьютерные вирусы, черви, руткиты и шпионское программное обеспечение, вредоносные программы представляют собой серьезную опасность для миллионов пользователей компьютеров, делая их уязвимыми, в частности, в отношении потери данных и секретной информации, хищения личных данных и снижения производительности.

Для обнаружения вредоносных программ, заражающих компьютерную систему пользователя и, кроме того, для удаления или остановки выполнения таких вредоносных программ может быть использовано программное обеспечение защиты данных. Из области техники известно несколько методов обнаружения вредоносных программ. Некоторые из этих методов основаны на сравнении фрагмента кода вредоносного объекта с библиотекой сигнатур, указывающих на вредоносность. Другие обычные методы позволяют обнаруживать набор поведенческих характеристик вредоносного объекта, указывающий на его вредоносность.

Программное обеспечение защиты данных может потребовать от компьютерной системы пользователя значительной затраты вычислительных ресурсов. Распространение вредоносных объектов ведет к постоянному усложнению программ обнаружения вредоносного программного обеспечения, баз данных сигнатур и методов эвристического анализа поведения, которые могут дополнительно замедлить операции по защите от вредоносных программ. Чтобы уменьшить вычислительные затраты, программное обеспечение защиты данных может содержать различные процедуры оптимизации, но обычно каждая такая процедура направлена на специальный случай или категорию вредоносных программ, и не может быть надлежащим образом применена к недавно обнаруженным вредоносным программам.

Чтобы соответствовать быстро изменяющемуся набору угроз, существует большая заинтересованность в разработке быстрых, надежных и настраиваемых решений по защите от вредоносных программ.

Раскрытие изобретения

Согласно одному аспекту, клиентская система содержит по меньшей мере один процессор, сконфигурированный для выполнения целевого процесса, причем целевой процесс содержит экземпляр главного исполняемого модуля и экземпляр общей библиотеки, причем по меньшей мере один процессор дополнительно сконфигурирован для приема от сервера первого индикатора репутации главного исполняемого модуля и второго индикатора репутации модуля общей библиотеки, причем первый индикатор репутации модуля определен в соответствии с поведением еще одного экземпляра главного исполняемого модуля. Сервер сконфигурирован для выполнения транзакций по защите от вредоносных программ с множеством клиентских систем, содержащих клиентскую систему. По меньшей мере один процессор дополнительно сконфигурирован для определения, в ответ на прием и второго индикатора репутации модуля, индикатора репутации целевого процесса в соответствии с первым и вторым индикатором репутации модуля, причем индикатор репутации процесса показывает, является ли целевой процесс, вероятно, вредоносным. По меньшей мере один процессор дополнительно сконфигурирован для конфигурирования, в ответ на определение индикатора репутации целевого процесса, сканирования на наличие вредоносных программ в соответствии с

индикатором репутации процесса, причем сканирование на наличие вредоносных программ выполняется посредством клиентской системы, чтобы определить, является ли целевой процесс вредоносным.

Согласно еще одному аспекту, сервер содержит по меньшей мере один процессор, сконфигурированный для определения первого индикатора репутации главного исполняемого модуля и второго индикатора репутации модуля общей библиотеки, причем первый индикатор репутации модуля определен в соответствии с поведением экземпляра главного исполняемого модуля. По меньшей мере один процессор дополнительно сконфигурирован для передачи, в ответ на определение первого индикатора репутации модуля и второго индикатора репутации модуля, первого и второго индикатора репутации модуля в одну клиентскую систему из множества клиентских систем, сконфигурированную для осуществления с сервером транзакций по защите от вредоносных программ. Клиентская система сконфигурирована для выполнения целевого процесса, причем целевой процесс содержит еще один экземпляр первой общей библиотеки и экземпляр второй общей библиотеки. Клиентская система дополнительно сконфигурирована для определения индикатора репутации целевого процесса в соответствии с первым и вторым индикатором репутации модуля, причем индикатор репутации процесса показывает, является ли целевой процесс, вероятно, вредоносным. Клиентская система дополнительно сконфигурирована для конфигурирования, в ответ на определение индикатора репутации, сканирования на наличие вредоносных программ в соответствии с индикатором репутации процесса, причем сканирование на наличие вредоносных программ выполняется посредством клиентской системы, чтобы определить, является ли целевой процесс вредоносным.

Согласно еще одному аспекту, постоянный машиночитаемый носитель хранит команды, которые в случае их выполнения конфигурируют по меньшей мере один процессор клиентской системы, выполняющий целевой процесс, причем целевой процесс содержит экземпляр главного исполняемого модуля и экземпляр общей библиотеки, чтобы принимать от сервера первый индикатор репутации главного исполняемого модуля и второй индикатор репутации общей библиотеки, причем первый индикатор репутации модуля определен в соответствии с поведением еще одного экземпляра главного исполняемого модуля. Сервер сконфигурирован для выполнения транзакций по защите от вредоносных программ со множеством клиентских систем, содержащих клиентскую систему. По меньшей мере один процессор дополнительно сконфигурирован для определения, в ответ на прием первого и второго индикатора репутации модуля, индикатора репутации целевого процесса в соответствии с первым и вторым индикатором репутации модуля, причем индикатор репутации процесса показывает, является ли целевой процесс, вероятно, вредоносным. По меньшей мере один процессор дополнительно сконфигурирован для конфигурирования, в ответ на определение индикатора репутации целевого процесса, сканирования на наличие вредоносных программ в соответствии с индикатором репутации процесса, причем сканирование на наличие вредоносных программ выполняется посредством клиентской системы, чтобы определить, является ли целевой процесс вредоносным.

Согласно еще одному аспекту, клиентская система содержит по меньшей мере один процессор, сконфигурированный для выполнения целевого процесса, причем целевой процесс содержит экземпляр первого исполняемого модуля и экземпляр второго исполняемого модуля. По меньшей мере один процессор дополнительно сконфигурирован для приема от сервера первого индикатора репутации модуля первого исполняемого модуля и второго индикатора репутации модуля второго исполняемого

модуля, причем первый индикатор репутации модуля определен в соответствии с поведением еще одного экземпляра первого исполняемого модуля. Сервер сконфигурирован для выполнения транзакций по защите от вредоносных программ с множеством клиентских систем, содержащих клиентскую систему. По меньшей мере один процессор дополнительно сконфигурирован для определения, в ответ на прием первого и второго индикатора репутации модуля, индикатора репутации целевого процесса в соответствии с первым и вторым индикатором репутации модуля, причем индикатор репутации процесса показывает, является ли целевой процесс, вероятно, вредоносным. По меньшей мере один процессор дополнительно сконфигурирован для конфигурирования, в ответ на определение индикатора репутации целевого процесса, сканирования на наличие вредоносных программ в соответствии с индикатором репутации процесса, причем сканирование на наличие вредоносных программ выполняется посредством клиентской системы, чтобы определить, является ли целевой процесс вредоносным.

Согласно еще одному аспекту, сервер содержит по меньшей мере один процессор, сконфигурированный для выполнения транзакций по защите от вредоносных программ с множеством клиентских систем, причем множество клиентских систем содержит клиентскую систему. Клиентская система сконфигурирована для выполнения целевого процесса, причем целевой процесс содержит экземпляр первого исполняемого модуля и экземпляр второго исполняемого модуля. По меньшей мере один процессор дополнительно сконфигурирован для определения первого индикатора репутации первого исполняемого модуля и второго индикатора репутации второго исполняемого модуля, причем первый индикатор репутации модуля определен в соответствии с поведением еще одного экземпляра первого исполняемого модуля, и сконфигурирован для передачи в клиентскую систему, в ответ на определение первого и второго индикатора репутации модуля, первого и второго индикатора репутации модуля. Клиентская система дополнительно сконфигурирована для определения индикатора репутации целевого процесса в соответствии с первым и вторым индикатором репутации модуля, причем индикатор репутации целевого процесса показывает, является ли целевой процесс, вероятно, вредоносным. Клиентская система дополнительно сконфигурирована для конфигурирования, в ответ на определение индикатора репутации, сканирования на наличие вредоносных программ в соответствии с индикатором репутации процесса, причем сканирование на наличие вредоносных программ выполняется посредством клиентской системы, чтобы определить, является ли целевой процесс вредоносным.

Краткое описание чертежей

Вышеупомянутые аспекты и преимущества настоящего изобретения могут быть лучше поняты при прочтении последующего подробного описания со ссылками на чертежи, на которых:

- на фиг. 1 показан пример системы защиты от вредоносных программ, содержащей множество клиентских систем и сервер репутации, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 2 показан пример изолированной среды, например, корпоративной интрасети, защищенной от вредоносных программ согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 3-А показан пример конфигурации аппаратных средств клиентской системы согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 3-В показан пример конфигурации аппаратных средств сервера репутации согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 4 показан пример набора программных объектов, содержащего приложение безопасности, защищающее клиентскую систему от вредоносных программ согласно некоторым вариантам осуществления настоящего изобретения;

5 - на фиг. 5 показан пример набора программных объектов, выполняющихся в клиентской системе согласно некоторым вариантам осуществления настоящего изобретения, причем объекты показаны с точки зрения уровней привилегий процессора;

- на фиг. 6 показана схема примера приложения безопасности, содержащего менеджер репутации и движок защиты от вредоносных программ, согласно некоторым вариантам осуществления настоящего изобретения;

10 - на фиг. 7 показан пример обмена данными между менеджером репутации и движком защиты от вредоносных программ, показанными на фиг. 6, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 8 показан пример менеджера репутации, принимающего данные из кэша репутации и сервера репутации, согласно некоторым вариантам осуществления настоящего изобретения;

15 - на фиг. 9 показаны два процесса, выполняющихся в клиентской системе, причем каждый процесс содержит множество исполняемых модулей; дополнительно, на фиг. 9 показан индикатор репутации процесса, определенный для каждого процесса, а также индикатор репутации модуля, определенный для каждого исполняемого модуля, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 10-А показан пример индикатора репутации облака согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 10-В показан пример индикатора репутации модуля согласно некоторым вариантам осуществления настоящего изобретения;

25 - на фиг. 10-С показан пример индикатора репутации процесса согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 11 показан пример последовательности этапов, выполняемых монитором активности, показанным на фиг. 7 и 8, согласно некоторым вариантам осуществления настоящего изобретения;

30 - на фиг. 12 показан пример последовательности этапов, выполняемых модулем принятия решения, показанным на фиг. 7 и 8, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 13 показан пример последовательности этапов, выполняемых модулем принятия решения после приема уведомления о запуске процесса, согласно некоторым вариантам осуществления настоящего изобретения;

35 - на фиг. 14 показан пример последовательности этапов, выполняемых локальным сервером репутации, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 15 показан пример последовательности этапов, выполняемых модулем принятия решения, показанным на фиг. 7 и 8, после приема уведомления системы безопасности, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 16 показан пример последовательности этапов, выполняемых движком защиты от вредоносных программ, показанным на фиг. 6 и 7, согласно некоторым вариантам осуществления настоящего изобретения;

40 - на фиг. 17 показан пример конфигурации системы обновления репутации, показанной на фиг. 1, согласно некоторым вариантам осуществления настоящего изобретения;

- на фиг. 18 показан пример последовательности этапов, выполняемых системой обновления репутации согласно некоторым вариантам осуществления настоящего

изобретения.

Осуществление изобретения

В нижеследующем описании подразумевается, что все перечисленные соединения между схемами и устройствами могут представлять собой непосредственные рабочие соединения или косвенные рабочие соединения посредством промежуточных схем и устройств. Набор элементов содержит один или более элементов. Любое упоминание элемента следует понимать, как относящееся по меньшей мере к одному элементу. Множество элементов содержит по меньшей мере два элемента. Если не указано другого требования, любые описанные этапы способа не обязательно должны выполняться в определенном показанном порядке. Первый элемент (например, данные), полученный из второго элемента, охватывает первый элемент, равный второму элементу, а также первый элемент, генерируемый посредством обработки второго элемента, и опционально другие данные. Принятие определения или решения в соответствии с параметром охватывает принятие определения или решения в соответствии с этим параметром и опционально в соответствии с другими данными. Если не дано другого определения, индикатор некоторого количества/данных может представлять собой само количество/данные, или индикатор, отличающийся от самого количества/данных. Если не дано другого определения, процесс представляет собой экземпляр компьютерной программы, причем компьютерная программа представляет собой последовательность команд, определяющих выполнение компьютерной системой специфической задачи. Если не дано другого определения, хэш представляет собой результат хэш-функции. Если не дано другого определения, хэш-функция представляет собой математическое преобразование соответствия последовательности символов переменной длины (например, знаков, битов) и данных фиксированной длины, например, числа или строки битов. Машиночитаемые носители охватывают постоянные носители, например, магнитные, оптические и полупроводниковые носители информации (например, жесткие диски, оптические диски, флеш-память, динамическую оперативную память), а также линии связи, например, токопроводящие кабели и волоконно-оптические линии связи. Согласно некоторым вариантам осуществления изобретения среди прочего в данном изобретении предложены компьютерные системы, содержащие аппаратные средства (например, один или более процессоров), запрограммированные для осуществления описанных здесь способов, а также содержащиеся на машиночитаемых носителях команды на машинном языке для осуществления описанных здесь способов.

Следующее описание на примерах, не обязательно носящих ограничительный характер, иллюстрирует варианты осуществления изобретения.

На фиг. 1 показан пример системы 5 защиты от вредоносных программ согласно некоторым вариантам осуществления настоящего изобретения. Система 5 содержит набор клиентских систем 10a-10c и центральный сервер 14 репутации, соединенные посредством сети 20 связи. Кроме того, центральный сервер репутации может быть соединен с центральной базой 16 данных репутации и системой 18 обновления репутации. В некоторых вариантах осуществления изобретения система 5 может дополнительно содержать набор изолированных сред 12a-12b (например, корпоративные интрасети), соединенных с сетью 20. Сеть 20 может представлять собой глобальную сеть, например, Интернет, тогда как части сети 20 также могут содержать локальную сеть (LAN).

Клиентские системы 10a-10c могут содержать компьютеры конечного пользователя, каждый из которых имеет процессор, память и устройство хранения данных. В этих компьютерах может работать операционная система, например, Windows®, MacOS® или Linux. Некоторые клиентские вычислительные системы 10a-10c могут представлять

собой мобильные вычислительные устройства и/или устройства дистанционной связи, например, планшетные персональные компьютеры, мобильные телефоны, персональные цифровые помощники (PDA), носимые компьютерные устройства, бытовые приборы, например, телевизоры или аудиоплееры, или любое другое электронное устройство, содержащее процессор и модуль памяти и нуждающееся в защите от вредоносных программ. В некоторых вариантах осуществления изобретения клиентские системы 10a-10c могут представлять отдельных клиентов, или несколько клиентских систем могут принадлежать одному и тому же клиенту.

В некоторых вариантах осуществления изобретения центральный сервер 14 репутации сконфигурирован для обработки данных репутации по запросу клиентских систем 10a-10c. База 16 данных репутации может содержать репозиторий данных репутации, например, индикаторов репутации модуля, определенных для множества исполняемых модулей, как подробно описано ниже. В одном из примеров осуществления изобретения сервер 14 может по запросу выбирать данные репутации из центральной базы 16 данных и передавать эти данные репутации в клиентские системы 10a-10c. В некоторых вариантах осуществления изобретения система 18 обновления репутации сконфигурирована для хранения и/или обновления данных репутации в базе 16 данных репутации, как описано ниже.

На фиг. 2 показана изолированная среда 12, например, корпоративная интрасеть, соединенная с сетью 20. Среда 12 содержит набор клиентских систем 10d-10e и локальный сервер 114 репутации, причем все они соединены с локальной сетью 120. Сеть 120 может представлять собой, например, локальную сеть. В некоторых вариантах осуществления изобретения изолированная среда 12 может дополнительно содержать кэш 22 сервера репутации и специфическую для этой среды базу 24 данных репутации, которые соединены с локальной сетью 120. Локальный сервер 114 репутации может быть сконфигурирован для обработки данных репутации по запросу клиентских систем 10d-10e, например, для выборки данных репутации из кэша 22 репутации и/или специфической для среды базы 24 данных репутации и передачи этих данных в запрашивающие клиентские системы 10d-10e. Кэш 22 и база 24 данных могут быть сконфигурированы для хранения данных репутации, например, индикаторов репутации модуля, как показано ниже. В некоторых вариантах осуществления изобретения локальный сервер 114 репутации может быть дополнительно сконфигурирован для обмена данными с центральным сервером 14 репутации, например, для выборки данных репутации из сервера 14 и хранения этих данных в кэше 22 и/или локальной базе 114 данных репутации.

На фиг. 3-А показан пример конфигурации аппаратных средств клиентской системы 10, например, клиентских систем 10a-10e, показанных на фиг. 1 и 2, выполняющих операции по защите от вредоносных программ согласно некоторым вариантам осуществления настоящего изобретения. Клиентская система 10 может представлять собой корпоративное вычислительное устройство, например, корпоративный сервер, или устройство конечного пользователя, например, персональный компьютер или смартфон. На фиг. 3 показан пример компьютерной системы, при этом другие клиентские системы, например, мобильные телефоны или планшеты, могут иметь другую конфигурацию. В некоторых вариантах осуществления изобретения система 10 содержит набор физических устройств, в том числе процессор 32, модуль 34 памяти, набор устройств 36 ввода, набор устройств 38 вывода, набор устройств 40 хранения данных и набор сетевых адаптеров 42, причем все они соединены посредством набора шин 44.

В некоторых вариантах осуществления изобретения процессор 32 содержит

физическое устройство (например, многоядерную интегральную микросхему), сконфигурированное для выполнения вычислительных и/или логических операций посредством набора сигналов и/или данных. В некоторых вариантах осуществления изобретения такие логические операции подают в процессор 32 в виде

5 последовательности процессорных команд (например, в виде машинного кода или другого типа программного обеспечения). Модуль 34 памяти может содержать постоянный машиночитаемый носитель (например, ОЗУ), хранящий данные/сигналы, полученные или сгенерированные процессором 32 в процессе выполнения команд. Устройства 36 ввода, помимо прочего, могут содержать компьютерные клавиатуры, 10 мышь и микрофоны, в том числе соответствующие аппаратные интерфейсы и/или адаптеры, позволяющие пользователю вводить данные и/или команды в клиентскую систему 10. Устройства 38 вывода, помимо прочего, могут содержать экраны дисплея и громкоговорители, а также аппаратные интерфейсы/адаптеры, например, графические платы, благодаря которым система 10 может передавать данные пользователю. В 15 некоторых вариантах осуществления изобретения устройства 36 ввода и устройства 38 вывода, например, в случае устройств с сенсорным экраном, могут совместно использовать общую часть программного обеспечения. Устройства 40 хранения данных содержат машиночитаемые носители, обеспечивающие возможность энергонезависимого хранения, чтения и записи команд программного обеспечения и/или данных. Примерные 20 устройства 40 хранения данных содержат магнитные и оптические диски, устройства на основе флеш-памяти, а также съемные носители, например, CD-диски и/или DVD-диски и приводы. Набор сетевых адаптеров 42 обеспечивает возможность подключения клиентской системы 10 к компьютерной сети, например, к сетям 20, 120, и/или другим устройствам/компьютерным системам. Вместе шины 44 представляют собой множество 25 систем, периферийных и микропроцессорных шин, и/или все другие схемы, обеспечивающие возможность обмена данными между устройствами 32-42 клиентской системы 10. Например, шины 44 могут содержать, помимо прочего, северный мост, соединяющий процессор 32 с памятью 34, и/или южный мост, соединяющий процессор 32 с устройствами 36-42.

30 На фиг. 3-В показан пример конфигурации аппаратных средств сервера репутации, например, центрального сервера 14 репутации, показанного на фиг. 1, или локального сервера 114 репутации, показанного на фиг. 2. Сервер 14 содержит процессор 132 сервера, память 134 сервера, набор запоминающих устройств 140 сервера и набор сетевых адаптеров 142, причем все они соединены посредством набора шин 144. Работа 35 устройств 132, 134, 140 и 142 может походить на работу устройств 32, 34, 40 и 42, описанных выше. Например, процессор 132 сервера может содержать физическое устройство, сконфигурированное для выполнения вычислительных и/или логических операций посредством набора сигналов и/или данных. Память 134 сервера может содержать постоянный машиночитаемый носитель (например, ОЗУ), хранящий данные/ 40 сигналы, полученные или сгенерированные процессором 132 в ходе выполнения вычислений. Сетевые адаптеры 142 обеспечивают возможность подключения клиентской системы 14 к компьютерной сети, например, к сетям 20, 120.

На фиг. 4 показан пример набора программных объектов, выполняющихся в клиентской системе 10, согласно некоторым вариантам осуществления настоящего 45 изобретения. Гостевая операционная система (ОС) 46 содержит программное обеспечение, обеспечивающее интерфейс для аппаратных средств клиентской системы 10, и действует в качестве хоста для набора программных приложений 52а-52с и 54. ОС 46 может содержать любую широкодоступную операционную систему, например,

Windows®, MacOS®, Linux®, iOS®, или Android™. Приложения 52a-52c, например, могут содержать приложение для обработки текстов, приложение для обработки изображений, приложение базы данных, браузерное приложение и приложение, использующее электронные средства связи. В некоторых вариантах осуществления изобретения приложение 54 безопасности сконфигурировано для осуществления операций по защите от вредоносных программ, как подробно описано ниже, чтобы защитить клиентскую систему 10 от вредоносных программ. Приложение 54 безопасности может представлять собой автономную программу, или оно может быть частью программного пакета, содержащего, помимо прочего, компоненты для защиты от вредоносных программ, спама и шпионского программного обеспечения.

На фиг. 6 показан набор программных объектов, выполняющихся в клиентской системе 10, показанный с точки зрения уровней привилегий процессора, которые также известны из области техники, как слои или кольца защиты. В некоторых вариантах осуществления изобретения каждый такой слой или кольцо защиты отличается набором команд, который разрешается выполнять программному объекту, выполняющемуся на соответствующем уровне привилегий процессора. Если программный объект пытается выполнить команду, выполнение которой на соответствующем уровне привилегий не разрешено, то эта попытка может вызвать событие в работе процессора, например, исключительное событие или событие ошибки. Большая часть компонентов операционной системы 46 выполняется на уровне привилегий процессора, известном из уровня техники как уровень ядра или режим ядра (например, в случае платформ Intel, кольцо 0). Приложение 52 выполняется с более низкой привилегией процессора, чем ОС 46 (например, кольцо 3 или режим пользователя).

В некоторых вариантах осуществления изобретения части приложения 54 безопасности могут выполняться на пользовательском уровне привилегий процессора, то есть на том же уровне, что и приложение 52. Например, такие части могут содержать графический интерфейс пользователя, информирующий пользователя о любых вредоносных программах или угрозах безопасности, обнаруженных в соответствующей VM, и прием вводимых пользователем данных, например, показывающих требуемую опцию конфигурации для приложения 54. Другие части приложения 54 могут выполняться на уровне привилегий ядра. Например, приложение 54 может установить драйвер 55 защиты от вредоносных программ, обеспечивающий для программы 54 защиты от вредоносного программного обеспечения функцию уровня ядра, например, для сканирования памяти на вредоносные сигнатуры и/или обнаружения вредоносного поведения процессов и/или других программных объектов, выполняющихся в ОС 46. В некоторых вариантах осуществления изобретения приложение 54 безопасности дополнительно содержит менеджер 58 репутации, части которого могут выполняться в режиме ядра.

На фиг. 6 показана схема примера приложения 54 безопасности согласно некоторым вариантам осуществления настоящего изобретения. Приложение 54 содержит движок 56 защиты от вредоносных программ и клиентский кэш 122 репутации, причем оба они соединены с менеджером 58 репутации. В некоторых вариантах осуществления изобретения клиентский кэш 122 репутации сконфигурирован для временного хранения данных репутации, например, индикаторов репутации модуля, и передачи этих данных в менеджер 58 репутации. В некоторых вариантах осуществления изобретения менеджер 58 репутации сконфигурирован для определения данных репутации, определенных для множества программных объектов, содержащих приложения, процессы и исполняемые модули, хранения и/или выборки этих данных из кэша 122 и передачи их в движок 56

защиты от вредоносных программ.

В некоторых вариантах осуществления изобретения движок 56 сконфигурирован для сканирования программных объектов, выполняющихся в клиентской системе 10, например, приложений 52а-52с, показанных на фиг. 4. Такое сканирование на наличие вредоносных программ может содержать определение того, содержит ли программный объект вредоносный код, и последующее удаление соответствующего кода или предотвращение выполнения соответствующего кода. В некоторых вариантах осуществления изобретения часть кода рассматривают, как вредоносную, если она сконфигурирована для осуществления любого действия из набора вредоносных действий. Такие вредоносные действия могут содержать любое действие, ведущее к утрате конфиденциальности или секретных данных, или к снижению производительности со стороны пользователя. Некоторые примеры содержат модификацию или удаление данных без ведома или санкции пользователя, а также изменение выполнения допустимых программ, выполняющихся в клиентской системе 10. Другие примеры вредоносных действий содержат получение секретных данных или персональных данных пользователя, например, паролей, деталей регистрации, данных кредитных карт или банковского счета, или конфиденциальных документов. Другие примеры вредоносных действий содержат несанкционированный перехват или подслушивание диалогов пользователя и/или обмена данными с третьей стороной. Другие примеры содержат применение клиентской системы 10 для передачи незатребованной информации (спама) и применение клиентской системы 10 для передачи вредоносных запросов на получение данных в удаленные компьютерные системы, например, атаки типа «отказ в обслуживании».

В некоторых вариантах осуществления изобретения целевые объекты, сканируемые движком 56, содержат выполняющиеся в пользовательском режиме приложения, процессы и исполняемые модули. Если не дано другого определения, процесс представляет собой экземпляр компьютерной программы, например, приложение или часть операционной системы 46, и отличается тем, что он имеет по меньшей мере поток выполняющихся задач и область виртуальной памяти, присвоенную ему операционной системой и содержащую исполняемый код. Если не дано другого определения, исполняемый модуль представляет собой компонент или стандартный блок процесса, причем каждый такой модуль содержит исполняемый код. Примеры исполняемых модулей, помимо прочего, содержат главный исполняемый файл процесса (например, в Windows® - файл с расширением ".exe") и общую библиотеку (например, динамически подключаемую библиотеку, DLL). В некоторых вариантах осуществления изобретения главный исполняемый модуль процесса содержит первую машинную команду, выполняющуюся при запуске соответствующего процесса. Библиотеки представляют собой автономные участки кода, выполняющие различные функциональные аспекты программы. Общие библиотеки могут быть использованы независимо друг от друга более чем одной программой. В клиентских системах 10, в которых выполняются такие операционные системы, как Linux® или MacOS®, могут быть идентифицированы исполняемые модули аналогичного типа. Исполняемые модули могут быть загружены и/или выгружены в память/из памяти во время запуска и/или выполнения соответствующего процесса. Чтобы осуществить сканирование на наличие вредоносных программ, движок 56 может применить любой известный в области техники способ защиты от вредоносных программ, например, сравнение сигнатур и сканирование поведения.

В некоторых вариантах осуществления изобретения, чтобы способствовать

обнаружению вредоносных программ, менеджер 58 репутации взаимодействует с движком 56 защиты от вредоносных программ. Например, менеджер 58 репутации может передавать в движок 56 защиты от вредоносных программ индикатор репутации

5 В некоторых вариантах осуществления изобретения репутация программного объекта показывает степень уверенности в том, что соответствующий объект не является вредоносным. Например, индикатор репутации может показывать, что соответствующий объект является надежным, ненадежным или неизвестным. В ответ движок 56 защиты от вредоносных программ может обеспечить для надежных объектов преференциальный

10 режим. Например, движок 56 может использовать для сканирования надежных объектов нестрогий протокол системы защиты, а для сканирования неизвестных или ненадежных объектов - строгий протокол, причем нестрогий протокол системы защиты требует меньших вычислительных затрат, чем строгий протокол. В одном таком примере нестрогий протокол системы защиты может дать команду движку 56 применить при

15 сканировании надежного объекта только поднабор способов определения вредоносных программ и/или только поднабор методов эвристического анализа, которые позволяют идентифицировать вредоносные программы, в то время как строгий протокол системы защиты могут использовать весь набор способов и/или методов эвристического анализа, доступных для движка 56. Как показано ниже, в некоторых вариантах осуществления

20 изобретения нестрогий протокол системы защиты может быть специально приспособлен для каждого процесса или исполняемого модуля. Например, менеджер 58 репутации может сконфигурировать движок 56 для применения специфического для процесса протокола, передав в движок 56 специфический для процесса набор значений параметров сканирования, применяемых для создания примера набора параметров конфигурации

25 движка 56 при сканировании или контроле соответствующего процесса. Таким образом, протокол сканирования/мониторинга может отличаться как для надежного и ненадежного объекта, так и для надежного объекта и другого надежного объекта.

В некоторых вариантах осуществления изобретения репутация целевого программного объекта может быть определена в соответствии с репутацией набора

30 стандартных блоков соответствующего процесса. Репутация стандартных блоков может быть сохранена в локальном репозитории (например, в кэшах 22, 122) и/или в удаленном репозитории (например, в центральной базе 16 данных) и повторно использоваться для каждого экземпляра соответствующих стандартных блоков. Напротив, репутация самого целевого объекта может быть определена динамически, то есть повторно

35 рассчитываться менеджером 58 репутации в ответ на событие, связанное с безопасностью и/или определенными событиями в жизненном цикле целевого объекта. Такой подход позволяет определить предпочтительную градацию или гранулярность сохраненных данных репутации. Размер и тип стандартных блоков может изменяться в зависимости от варианта осуществления изобретения. В некоторых вариантах осуществления

40 изобретения стандартные блоки представляют собой исполняемые модули (например, главные исполняемые файлы и общие библиотеки). Для упрощения при дальнейшем обсуждении будут описаны только такие применения. Другие примеры стандартных блоков, помимо прочего, включают исполняемые скрипты, вызываемые соответствующим процессом (например, скрипты Perl, Visual Basic, и Python), и

45 интерпретированные файлы (например, файлы Java® JAR). Специалист в области техники поймет, что описанные здесь системы и способы могут быть перенесены на стандартные блоки других типов и другие уровни гранулярности.

На фиг. 7 показан еще один менеджер 58 репутации и движок 56 защиты от

вредоносных программ, а также пример обмена данными между компонентами 56 и 58 в некоторых вариантах осуществления настоящего изобретения. Движок 56 защиты от вредоносных программ может содержать множество элементов 76a-76d защиты. В некоторых вариантах осуществления изобретения элементы 76a-76d представляют собой компоненты движка 56, которые могут выполняться независимо друг от друга. Примеры элементов защиты содержат отдельные компоненты защиты от вредоносных программ, каждый из которых применяет особый способ, в том числе межсетевой экран, сканер поведения и движок сравнения сигнатур. В некоторых вариантах осуществления изобретения межсетевой экран перехватывает входящий и/или исходящий сетевой трафик, относящийся к программным объектам, выполняющимся в клиентской системе 10, и может допустить, отклонить или модифицировать прохождение или содержание такого трафика в соответствии с указанными пользователем правилами и/или внутренним методов эвристического анализа. В некоторых вариантах осуществления изобретения сканер поведения осуществляет мониторинг поведения (например, обнаруживает такие действия, как запись в дисковый файл или редактирование раздела реестра операционной системы) программных объектов, выполняющихся в клиентской системе 10. Движок сравнения сигнатур может делать попытки, сравнить кодовую последовательность программного объекта, выполняющегося в клиентской системе 10, с перечнем кодовых последовательностей, указывающих на вредоносность, которые известны как сигнатуры. Это сравнение может показать, что соответствующий объект содержит вредоносное программное обеспечение. Еще один пример элемента защиты представляет собой защитную процедуру, которая может объединять несколько способов или компонентов защиты от вредоносных программ, например, «применение способа А и, если результат равен 1, применение способа В, и так далее». Другие примеры защитных процедур содержат сканирование по запросу и сканирование при доступе. Еще один пример элемента защиты представляет собой набор методов эвристического анализа для идентификации вредоносных программ.

В некоторых вариантах осуществления изобретения элементы 76a-76d защиты могут быть сконфигурированы с использованием набора параметров сканирования, причем изменение значения одного из параметров сканирования может изменить работу соответствующего элемента. Один из примеров параметра сканирования представляет собой пороговое значение, применяемое для сравнения с оценкой, указывающей на вредоносность. Если эта оценка превышает соответствующее пороговое значение, то процесс или модуль может рассматриваться, как вредоносный. Еще один пример параметра сканирования представляет собой набор методов эвристического анализа, применяемый сканером поведения при решении о том, является ли вредоносным процесс или модуль. Еще один пример параметра сканирования представляет собой набор нейронных весовых коэффициентов, применяемых для калибровки классификаторов нейронных сетей, чтобы сделать различие между неопасными и вредоносными объектами. Как подробно показано ниже, в некоторых вариантах осуществления изобретения благодаря передаче в движок 56 набора значений таких параметров сканирования менеджер 58 репутации может эффективным образом конфигурировать движок 56 защиты от вредоносных программ.

В некоторых вариантах осуществления изобретения менеджер 58 репутации (фиг. 7) содержит модуль 72 принятия решения, монитор 74 активности, соединенный с модулем 72 принятия решений, а также менеджер 68 облака и менеджер кэша 70, которые оба соединены с модулем 72 принятия решения. Менеджер 58 репутации может быть сконфигурирован для приема от движка 56 защиты от вредоносных программ запроса

60 репутации и/или уведомления 64 о событии, связанном с безопасностью, и передачи в движок 56 защиты от вредоносных программ индикатора 82 репутации процесса и/или предупреждения 66 системы защиты. Детали обмена такими данными приведены ниже.

5 В некоторых вариантах осуществления изобретения монитор 74 активности сконфигурирован для обнаружения событий в жизненном цикле программных объектов, например, приложений и процессов, выполняющихся в клиентской системе 10, чтобы
10 получить перечень таких контролируемых объектов и передавать эти события в модуль 72 принятия решения. Монитор 74, например, может обнаружить запуск и/или завершение приложений, процессов и/или потоков. Другие примеры событий, перехватываемых монитором 74, содержат процесс, загружающий или выгружающий
15 исполняемый модуль, например, DLL, из его пространства памяти или в его пространство памяти. Кроме того, монитор 74 активности может определять межобъектные отношения, например, какой исполняемый модуль запущен и каким процессом. Другие такие связи содержат отношения родства, например, отношения
20 между порождающим и порожденным процессом. В некоторых вариантах осуществления изобретения монитор 74 дополнительно может определять, внедрил ли выбранный объект код в другой объект, или является ли выбранный объект целью такого внедрения посредством другого программного объекта. Порожденный объект представляет собой
25 исполняемый объект, созданный другим объектом, который называют порождающим объектом, причем порожденный объект выполняется независимо от порождающего объекта. Примеры порожденных объектов представляют собой порожденные процессы, например, созданные посредством функции CreateProcess операционной системы Windows®, или посредством механизма ветвления в ОС Linux®. Внедрение кода является
30 общим названием, применяемым для обозначения ряда способов внедрения кодовой последовательности, например, динамически подключаемой библиотеки (DLL), в пространство памяти существующего процесса, чтобы изменить первоначальную функцию соответствующего процесса. Чтобы выполнить такие задачи, как обнаружение
35 запуска процесса и/или обнаружение внедрения кода, монитор 74 может применить любой способ, известный в области техники, например, вызов определенных функций ОС или перехват обращений к определенным функциям ОС. Например, в системе, работающей на ОС Windows®, монитор 74 может перехватывать обращение к функции LoadLibrary или функции CreateFileMapping, чтобы определить загрузку исполняемого модуля. В еще одном примере монитор 74 может регистрировать обратный вызов PsSetCreateProcessNotifyRoutine, чтобы определить запуск нового процесса, и/или он может перехватить обращение к функции CreateRemoteThread, чтобы определить выполнение
внедренного кода.

В некоторых вариантах осуществления изобретения модуль 72 принятия решения сконфигурирован для приема данных от монитора 74 активности, показывающих
40 появление события в жизненном цикле целевого процесса, например, запуска целевого процесса. Модуль 72 принятия решения может быть дополнительно сконфигурирован для запроса данных, показывающих репутацию исполняемого модуля целевого процесса (например, главного исполняемого файла или общей библиотеки, загруженной целевым процессом), у менеджера 70 кэша и/или менеджера 68 облака. Модуль 72 может быть
45 дополнительно сконфигурирован для определения репутации целевого процесса в соответствии с репутацией соответствующего исполняемого модуля и передачи индикатора 82, показывающего движку 56 защиты от вредоносных программ репутацию целевого процесса.

В некоторых вариантах осуществления изобретения модуль 72 принятия решения может быть сконфигурирован для приема от движка 56 защиты от вредоносных программ уведомления 64 о событии, связанного с безопасностью, указывающего на событие, связанное с безопасностью, вызванное выполнением целевого процесса или модуля, или происходящее во время выполнения целевого процесса или модуля. Такие события, связанные с безопасностью, обнаруживаются движком 56 защиты от вредоносных программ. Некоторые примеры содержат осуществление целевым процессом или модулем определенного действия, например, загрузку файла из интернета или модификацию раздела реестра ОС 46. Такие действия, осуществляемые целевым процессом или модулем, могут указывать на вредоносность сами по себе, или эти действия могут указывать на вредоносность, если они происходят в сочетании с другими действиями, осуществляемыми целевым процессом или модулем, или процессом, относящимся к целевому процессу (например, дочерним процессом целевого процесса). Модуль 72 принятия решения может быть дополнительно сконфигурирован для расчета (повторного расчета), в ответ на прием сообщения 64, индикатора 82 репутации целевого процесса в соответствии с сообщением 64 и передачи индикатора 82 в движок 56 защиты от вредоносных программ.

В некоторых вариантах осуществления изобретения модуль 72 принятия решения может быть сконфигурирован для анализа уведомления 64, и в случае определенных типов событий, связанных с безопасностью и показанных посредством сообщения 64, передачи в движок 56 защиты от вредоносных программ предупреждения 66 системы защиты. Предупреждение 66 системы защиты может показать движку 56 защиты от вредоносных программ, что соответствующая клиентская система 10 может быть заражена вредоносным программным обеспечением, и может сконфигурировать движок 56 для выполнения строгого протокола системы защиты, именуемого чрезмерно строгим режимом. Более подробное описание работы модуля 72 принятия решения дано ниже.

Менеджер 70 кэша и менеджер 68 облака могут быть сконфигурированы для выборки, по запросу модуля 72 принятия решения, данных репутации из кэша 222 репутации и сервера 214 репутации, соответственно. Кэш 222 может представлять собой, например, клиентский кэш 122 репутации (фиг. 6). В зависимости от применения, сервер 214 репутации может представлять собой центральный сервер 14 репутации (фиг. 1) или локальный сервер 114 репутации (фиг. 2). На фиг. 8 показан менеджер 70 кэша, принимающий из кэша 222 индикатор 80 репутации модуля, и менеджер 68 облака, принимающий от сервера 214 репутации индикатор 78 репутации облака, согласно некоторым вариантам осуществления настоящего изобретения. Менеджер 70 кэша может быть дополнительно сконфигурирован для хранения данных репутации, принятых менеджером 68 облака, в кэше 222 в течение разного, но ограниченного отрезка времени, а время жизни данных репутации в кэше может быть определено в соответствии с индикатором 78 облака.

На фиг. 9 показаны два примера процесса 84а, 84b и множество индикаторов репутации, согласно некоторым вариантам осуществления настоящего изобретения. Процесс 84а содержит исполняемые модули 86а-86с, например, главный исполняемый файл 86а и две библиотеки 86b-86с динамических связей. Процесс 84b включает главный исполняемый файл 86d и три динамически подключаемых библиотеки, две из которых являются экземплярами библиотек 86b-86с процесса 84а. Набор индикаторов 80а-80е репутации модуля определена для модулей 86а-86е соответственно, а набор индикаторов 82а, 82b репутации процесса определена для процессов 84а-84b соответственно.

В некоторых вариантах осуществления изобретения индикаторы 80а-80е репутации

модуля представляют собой элементы статических данных, сохраняемых и/или выбираемых из кэша 222 репутации и/или сервера 214 репутации (фиг. 8). Если два или большее число процессов загружают экземпляры одного и того же модуля, например, общие библиотеки (фиг. 9), то индикаторы репутации модуля могут быть идентичными для всех этих экземпляров. Индикатор 82а репутации процесса определен в соответствии с индикаторами 86а-86с репутации модуля, а индикатор 82b репутации процесса определен в соответствии с индикаторами 86b-86е репутации модуля. В отличие от индикаторов репутации модуля индикаторы 82а-82b репутации процесса могут быть динамически изменяемыми, повторно рассчитываемыми менеджером 58 репутации в ответ на события, связанные с безопасностью, происходящие в клиентской системе 10, и/или соответственно в ответ на события в цикле жизни процессов 84а, 84b.

Некоторые примеры форматов индикаторов репутации облака, модуля и процесса показаны соответственно на фиг. 10-А-10-С. Индикатор 78 репутации облака, помимо прочего, содержит набор значений 79 параметра сканирования облака и индикатор 79b истечения срока кэша. Индикатор 79b истечения срока кэша показывает время жизни индикатора 78 репутации облака, например, отрезок времени, в течение которого соответствующие данные репутации должны храниться в кэше 222. Это время жизни может изменяться в зависимости от исполняемых моделей. Благодаря точному определению времени жизни данных репутации в кэше некоторые варианты осуществления изобретения эффективным образом вынуждают обновление этих данных, принимаемых от сервера 214 репутации, и, следовательно, содержащих информацию о распространенности потенциального заражения соответствующего модуля.

В некоторых вариантах осуществления изобретения значения 79а параметров сканирования облака содержат набор значений параметров, применяемых для конфигурирования элементов 76а-76d защиты движка 56 защиты от вредоносных программ (фиг. 7) при сканировании/мониторинге соответствующего исполняемого модуля. В одном таком примере элемент F_1 защиты представляет собой поведенческий сканер, а значения параметра сканирования для F_1 включают кортеж $\{v_1, v_3, \dots\}$, причем значение v_1 параметра означает «активацию способа X сканирования поведения», параметр v_3 означает «блокировку набора Y методов эвристического анализа» и так далее. При мониторинге соответствующего исполняемого модуля данный пример элемента поведенческого сканера движка 56 получает команду на применение технологии X, в то же время блокируя набор Y методов эвристического анализа.

Некоторые варианты осуществления индикатора 80 репутации модуля (фиг. 10-В), помимо прочего, содержат набор значений 81а параметров сканирования модуля, индикатор 81b истечения срока кэша и индикатор 81с оценки модуля. Значения 81а параметров сканирования модуля показывают значения параметров, конфигурирующие движок 56 защиты от вредоносных программ для контроля и/или сканирования соответствующего модуля. Индикатор 81b истечения срока кэша показывает время жизни соответствующих данных репутации. В некоторых вариантах осуществления изобретения индикатор 81b истечения срока кэша идентичен индикатору 79b истечения срока кэша соответствующего модуля. Значения 81а параметров сканирования модуля могут быть идентичными значениям 79а параметров сканирования облака соответствующего модуля, или они могут отличаться от значений 79а после события, связанного с безопасностью, например, после осуществления соответствующим модулем действия, указывающего на вредоносность (см. ниже).

Индикатор 81с оценки модуля обеспечивает оценку степени доверия, присвоенной

в текущий момент времени соответствующему модулю. В некоторых вариантах осуществления изобретения индикатор 81с оценки модуля в соответствии с текущей информацией помечает соответствующий модуль, как надежный, ненадежный или неизвестный. Метка «надежный» может показывать, что соответствующий модуль, вероятно, не является вредоносным. Метка «ненадежный» может показывать, что соответствующий модуль подозрителен в отношении вредоносности (например, если соответствующий модуль осуществил действие, указывающее на вредоносность, например, внедрение кода в другой модуль или процесс). Метка «неизвестный» может показывать, что кэш 222 и/или сервер 214 не имеет данных о репутации соответствующего модуля.

В некоторых вариантах осуществления изобретения индикатор 82 репутации процесса (фиг. 10-С), помимо прочего, содержит набор значений 83а параметров сканирования процесса и индикатор 83с оценки процесса. Индикатор 83с оценки процесса показывает степень доверия, присвоенную в текущий момент времени соответствующему процессу. Примеры значений индикатора 83с содержат значение «позитивный», «негативный» и «нейтральный». Метка «позитивный» может показывать, что соответствующий процесс, вероятно, не является вредоносным, и она может дать команду движку 56 защиты от вредоносных программ на сканирование и/или мониторинг соответствующего процесса с применением значений 83а параметров сканирования. Метка «негативный» может указывать на подозрение во вредоносности и давать команду движку 56 защиты от вредоносных программ на анализ соответствующего процесса с применением заранее установленного строгого протокола системы защиты. Метка «нейтральный» может указывать на то, что соответствующий процесс не является ни совершенно надежным, ни ненадежным, и может дать команду движку 56 защиты от вредоносных программ на анализ соответствующего процесса с применением заранее установленного протокола системы защиты по умолчанию. В некоторых вариантах осуществления изобретения, если целевой процесс содержит набор исполняемых модулей, индикаторы 83а целевого процесса определены в соответствии с индикаторами 81а каждого набора исполняемых модулей целевого процесса, как описано ниже. Кроме того, как показано ниже, индикатор 83с оценки целевого процесса может быть определен в соответствии с индикаторами 81с оценки набора исполняемых модулей целевого процесса.

На фиг. 11 показан пример последовательности этапов, выполняемых монитором 74 активности (фиг. 7) согласно некоторым вариантам осуществления настоящего изобретения. В ходе последовательности этапов 302-304 монитор 74 ждет появления в клиентской системе 10 события-триггера. Пример события-триггера содержит события, происходящие на протяжении жизненного цикла процессов, выполняющихся в клиентской системе 10, например, запуск или завершение процесса. Другие примеры содержат загрузку или выгрузку процессом исполняемого модуля, например, DLL. Еще одним примером события-триггера является порождение порождающим процессом набора порождаемых процессов. Чтобы обнаружить появление таких событий-триггеров, монитор 74 активности может применить любой способ, известный из области техники, например, вызов определенных функций ОС или перехват обращений к определенным функциям ОС, например, функциям, используемым ОС 46 для запуска выполнения процесса.

На этапе 306 монитор 74 активности может идентифицировать тип события, например, запуск выполнения нового процесса. Если действительно происходит событие-триггер, то этап 308 может идентифицировать процесс (процессы), которые либо являются причиной соответствующего события-триггера, либо находятся под его воздействием.

В некоторых вариантах осуществления изобретения монитор 74 может определять идентичность таких процессов по структуре данных, используемых ОС 46 для представления каждого выполняющегося в данный момент времени процесса. Например, в случае Windows каждый процесс представляется, как блок процесса (EPROCESS), который, помимо прочего, содержит дескрипторы для каждого потока соответствующего процесса и уникальный идентификационный номер процесса, благодаря которому ОС 46 может идентифицировать соответствующий процесс из множества выполняющихся процессов. Для других операционных систем, например, Linux®, имеются аналогичные представления процесса. Если событие-триггер воздействует более чем на один процесс, то этап 308, кроме того, может содержать определение отношений между соответствующими процессами. Например, если порождающий процесс запускает порождаемый процесс, то монитор 74 может зарегистрировать идентичность порождающего и порождаемого процессов, а также тип их отношений (отношений родства). Затем на этапе 310 монитор 74 активности может передать в модуль 72 принятия решения уведомление, относящееся к этому событию-триггеру.

На фиг. 12 показан пример последовательности этапов, выполняемых модулем 72 принятия решения в соответствии с некоторыми вариантами осуществления настоящего изобретения. В ходе последовательности этапов 312-314 модуль 72 принятия решения ждет приема уведомления. Такие уведомления могут передаваться либо монитором 74 активности (как описано выше), либо движком 56 защиты от вредоносных программ (например, уведомление 64 о событии, связанном с безопасностью, показанное на фиг. 7). В случае приема уведомления на этапе 316 определяют, является ли это уведомление уведомлением о событии, связанном с безопасностью, полученным от движка 56 защиты от вредоносных программ, и если это так, на этапе 320 модуль 72 принятия решения действует в соответствии с типом принятого уведомления о событии, связанном с безопасностью (более подробно это описано ниже). Если уведомление не является уведомлением о событии, связанном с безопасностью, то на этапе 318 определяют, является ли это уведомление уведомлением о событии-триггере, принятым от монитора 74 активности, и если это так, на этапе 322 модуль 72 принятия решения действует в соответствии с типом события-триггера, переданного посредством данного уведомления.

На фиг. 13 показан пример последовательности этапов, выполняемых модулем 72 принятия решения на этапе 322 (фиг. 12), если событие-триггер содержит запуск процесса. На этапе 324 модуль 72 принятия решения может определить набор исполняемых модулей, загружаемых соответствующим процессом. Для выполнения этапа 324 модуль 72 принятия решения может применить любой известный из области техники способ. Например, в клиентских системах, в которых работает ОС Windows®, при запуске процесса ОС 46 устанавливает специфичную для процесса структуру данных, известных под названием «блок операционного окружения процесса» (PEB). Этот блок содержит данные, применяемые ОС 46 для управления ресурсами, связанными с соответствующим процессом. Такие данные содержат адреса виртуальной памяти загруженных модулей. Следовательно, модуль 72 принятия решения может идентифицировать загруженные модули в соответствии с PEB соответствующего процесса.

Далее, модуль 72 принятия решения может для каждого модуля, загружаемого целевым процессом, выполнить в цикле последовательность этапов 326-342. На этапе 326 выбирают загружаемый модуль. На этапе 328 модуль принятия решения рассчитывает индикатор идентификации выбранного модуля, причем, благодаря этому индикатору идентификации, кэш 222 и/или сервер 214 может однозначно

идентифицировать выбранный модуль. В некоторых вариантах осуществления изобретения индикатор идентификации, помимо прочего, содержит хэш участка кода выбранного модуля или участок кода выбранного модуля. Примеры хэш-функций, применяемых для расчета этого хэша, содержат защищенный алгоритм хеширования (SHA) и алгоритм выборки сообщений (MD). В некоторых вариантах осуществления изобретения менеджер 70 кэша может использовать идентифицирующий хэш в качестве ключа для поиска в кэше 222. Другие примеры индикаторов идентификации помимо прочего, могут содержать метаданные соответствующего модуля, например, имя файла, размер, версию и метку времени выбранного модуля.

На этапе 330 менеджеру 70 кэша выдается команда выбрать из кэша 222 данные репутации выбранного модуля. Этап 330 может содержать передачу в менеджер 70 кэша индикатора идентификации выбранного модуля и прием из менеджера 70 кэша индикатора 80 репутации выбранного модуля. На этапе 332 определяют, был ли успешным просмотр кэша, то есть были ли найдены в кэше 222 данные репутации выбранного модуля. Если эти данные были найдены, модуль принятия решения переходит к этапу 342, описанному ниже. Если в текущий момент запрошенных данных репутации в кэше не имеется (например, если срок действия запрошенных данных в кэше 222 истек), то на этапе 334 модуль 72 принятия решения может дать менеджеру 68 облака команду на поиск данных репутации на сервере 214 репутации. Этап 334 может содержать передачу на сервер 214 запроса репутации, содержащего индикатор идентификации (например, индикатор хэша) выбранного модуля и прием от сервера 214 индикатора 78 репутации облака. Чтобы получить индикатор 78, в некоторых вариантах осуществления сервера 214 может применяться идентифицирующий хэш выбранного модуля в качестве ключа поиска в центральной базе 16 репутации и/или базе 24 данных репутации, специфической для данной среды. Если такой поиск в базе заканчивается неудачей, то есть, если данные репутации выбранного модуля в базе (базах) данных репутации найдены не были, то сервер 214 может выдать индикатор ошибки.

В последовательности этапов 336-338 модуль 72 принятия решения может составить индикатор 80 репутации выбранного модуля в соответствии с индикатором 78 репутации облака выбранного модуля. В некоторых вариантах осуществления изобретения определение значений 81a параметров сканирования и индикатора 81b истечения срока кэша содержит копирование соответствующих значений из индикатора 78 репутации облака (элементы 79a и 79b соответственно, показанные на фиг. 10-A и 10-B). В некоторых вариантах осуществления изобретения на этапе 338 модуль 72 принятия решения может определить индикатор 81c оценки модуля, как надежный, ненадежный или неизвестный. В одном из примеров выбранный модуль признается надежным, если поиск в базе был успешным, то есть, если на этапе 334 выдан индикатор репутации облака для выбранного модуля. Если данные репутации выбранного модуля в базе (базах) репутации найдены не были, то выбранный модуль может получить оценку «неизвестный» («нейтральный»). В некоторых вариантах осуществления изобретения, если модуль 72 принятия решения принял от движка 56 защиты от вредоносных программ уведомление системы безопасности, показывающее, что существует подозрение, что выбранный модуль (или отдельный экземпляр выбранного модуля) является вредоносным, выбранный модуль может получить оценку «ненадежный», независимо от того, выдан ли на этапе 334 индикатор репутации облака для выбранного модуля, или нет.

На этапе 340 модуль 72 принятия решения дает менеджеру 70 кэша команду на

хранение в кэше 222 данных репутации выбранного модуля (например, индикатора 80). В некоторых вариантах осуществления изобретения, если существует подозрение во вредоносности соответствующего модуля, менеджер кэша может изменить значение индикатора 81b срока кэша. Например, если модуль принятия решения принял уведомление системы защиты, касающееся соответствующего модуля, то индикатор 81b может быть изменен, чтобы он показывал непрерывную запись (без времени истечения срока).

Затем на этапе 342 определяют, существуют ли еще подлежащие обработке исполняемые модули и, если такие модули имеются, выполнение программы продолжается с этапа 326, описанного выше. Если был обработан последний исполняемый модуль целевого процесса, то последовательность этапов 344-346 определяет индикатор 82 репутации целевого процесса в соответствии с индикаторами 80 репутации исполняемых модулей, составляющих целевой процесс. Пример таблицы решений для определения индикатора 82 репутации процесса приведен в таблице 1.

ТАБЛИЦА 1

Индикаторы оценки модуля	Индикаторы оценки процесса
Все Позитивные/Надежные	Позитивный/ Надежный
По меньшей мере один Нейтральный/Неизвестный, Все остальные Позитивные/ Надежные	Нейтральный/ Неизвестный
По меньшей мере один	Негативный/ Ненадежный
Негативный/Ненадежный	

На этапе 346 модуль 72 принятия решения определяет значения 83a параметров сканирования целевого процесса в соответствии со значениями 81a параметров сканирования исполняемых модулей, составляющих целевой процесс. В некоторых вариантах осуществления изобретения для каждого элемента 76a-76d защиты (фиг. 7) определяют соответствующий набор значений 83a параметров сканирования процесса в соответствии с установленной операцией над наборами, применяемой к наборам 81a значений параметров сканирования исполняемых модулей, составляющих целевой процесс, причем наборы 81a значений параметров соответствуют соответствующему элементу. Пример операций над наборами, применяемые для определения значений 83a параметров сканирования процесса, помимо прочего, содержат объединение и пересечение. В альтернативном варианте, если наборы 81a параметров сканирования модуля содержат двоичные значения, то операции над множествами могут содержать логические операторы И и ИЛИ. Ниже дан пример такого определения значений 83a параметров сканирования для процесса 84b, показанного на фиг. 9. Таблица 2 содержит примеры значений 81a параметров сканирования каждого модуля, составляющего процесс 84b и значений 83a параметров сканирования результирующего процесса 84b.

ТАБЛИЦА 2

Тип сканируемого параметра	Имя модуля	Значения параметра сканирования, элемент F_1		Значения параметра сканирования, элемент F_2	
		Пересечение/И	Объединение/ИЛИ	Пересечение/И	Объединение/ИЛИ
Модуль	B.exe	{a ₂ , a ₄ }	N/A	{b ₁ }	N/A
Модуль	X.dll	{a ₄ }	{a ₁ }	{b ₁ }	N/A

Модуль	Y.dll	{a ₂ , a ₄ }	N/A	{b ₁ }	{b ₃ }
Модуль	Z.dll	{a ₂ , a ₃ , a ₄ }	N/A	{b ₁ , b ₂ }	N/A
Процесс	B	{a ₄ }	{a ₁ }	{b ₁ }	{b ₃ }

В примере, приведенном в таблице 2, третий и пятый столбец содержат перечень наборов значений параметров, обрабатываемых с применением операции пересечения/И, тогда как четвертый и шестой столбец содержат перечень наборов значений параметров, обрабатываемых с применением операции объединения/ИЛИ. Элемент F_1 может представлять собой процедуру защиты в реальном времени, элемент F_2 может представлять собой поведенческий сканер. Значения параметров сканирования, например, могут означать следующее:

- a₁: активировать антивирусную технологию T₁;
- a₂: не выполнять мониторинг попыток ЧТЕНИЯ дискового файла;
- a₃: не выполнять мониторинг попыток ЗАПИСИ в дисковый файл;
- a₄: сканировать только исполняемые файлы;
- b₁: блокировать набор H₁ методов эвристического анализа;
- b₂: блокировать набор H₂ методов эвристического анализа;
- b₃: активировать технологию T₂ поведенческого сканирования.

В некоторых вариантах осуществления изобретения модуль 72 принятия решения может получить пересечение множеств {a₂, a₄}, {a₄}, {a₂, a₄}, и {a₂, a₃, a₄}, чтобы вывести {a₄}, и объединение множеств {a₁} и {N/A}, чтобы вывести {a₁}. Модуль 72 принятия решения дополнительно может получить объединение двух результирующих множеств {a₁} и {a₄}, чтобы вывести {a₁, a₄} в качестве набора значений 83а параметров сканирования процесса В, соответствующего элементу F_1 защиты. Аналогично в отношении элемента F_2 защиты модуль 72 принятия решения может получить пересечение множеств {b₁}, {b₁}, {b₁}, и {b₁, b₂}, чтобы вывести {b₁}, и объединение множеств {b₃} и {N/A}, чтобы вывести {b₃}. Таким образом, множество {b₁, b₃}, определенное как объединение результирующих множеств {b₁} и {b₃}, может представлять собой набор значений 83а параметров сканирования процесса В,

соответствующий элементу F_2 защиты. Используя вышеперечисленные значения различных параметров, значения 83а параметров сканирования процесса дают движку 56 защиты от вредоносных программ такую команду, что в процессе сканирования В компонент защиты в реальном времени сканирует только исполняемые объекты и активизирует антивирусную технологию T_1 , а компонент поведенческого сканера в то же время блокирует набор H_1 методов эвристического анализа и активизирует технологию T_2 поведенческого сканирования.

После расчета значений 83а параметров сканирования процесса и индикатора 83с оценки процесса модуль принятия решения может ассемблировать индикатор 82 репутации целевого процесса и в процессе операции 348 передать индикатор 82 в движок 56 защиты от вредоносных программ.

На фиг. 14 показан пример обработки запроса репутации в изолированной среде, например, в средах 12а, 12b (фиг. 1). По сравнению с примером системы, изображенной на фиг. 7 и 8, система, показанная на фиг. 14, вводит между центральным сервером 14 репутации и конечными клиентами 10 дополнительный уровень кэширования и опроса сервера. Благодаря уменьшению потока данных между сервером 14 и оконечными клиентскими системами 10 такие конфигурации могут ускорять операции по защите от вредоносных программ.

Кроме того, такие конфигурации, как изображенная на фиг. 2 и 14, делают возможной обработку данных репутации специфическим для данной среды способом. В некоторых вариантах осуществления изобретения специфическая для данной среды база 24 данных содержит набор индикаторов репутации модуля и/или облака, специально разработанный для соответствующей изолированной среды 12. В одном из таких примеров в клиентских системах 10d-10e (фиг. 2) корпоративной интрасети выполняется широко применяемое программное приложение X, например, Microsoft Office®. Приложение X загружает исполняемый модуль Y, уязвимый для вредоносных программ, поскольку соответствующая клиентская система подключена к Интернету. Если локальная сеть 120, соединяющая клиентские системы 10d-10e, соединена с Интернетом не непосредственно, например, если сеть 120 защищена при помощи межсетевого экрана, то приложение X уже не страдает от уязвимости из-за возможности соединения с Интернетом. Следовательно, в системах 10d-10e (то есть в изолированной среде 12) мониторинг приложения X в отношении такой уязвимости может не потребоваться, в то же время в системах, непосредственно соединенных с Интернетом, такой мониторинг может иметь большое значение. В некоторых вариантах осуществления изобретения в зависимости от того, как выполняется движок 56 в соответствующей клиентской системе - в изолированной среде 12 или вне изолированной среды 12, - могут быть установлены разные значения параметров сканирования, конфигурирующих движок 56 защиты от вредоносных программ. Таким образом, движок 56 может быть выполнен с возможностью сканирования или мониторинга экземпляра модуля Y, выполняющегося в среде 12, таким путем, который отличается от способа сканирования или контроля модуля Y, выполняющегося вне среды 12.

В еще одном примере специфики, зависящей от среды, фирма использует проприетарное программное приложение X, не применяемое другими клиентскими системами вне изолированной среды 12. Следовательно, данные репутации, относящиеся к исполняемому модулю Y приложения X, вероятно, не будут использоваться другими клиентскими системами. В некоторых вариантах осуществления изобретения такие данные репутации хранятся только в специфической для среды базе 24 данных репутации,

а не в центральной базе 16 данных репутации. Такие конфигурации могут повысить эффективность поиска в базе данных как для клиентов, работающих вне изолированной среды 12, так и для клиентов, работающих внутри среды 12.

В некоторых вариантах осуществления изобретения поток выполнения, показанный на фиг. 14, соответствует выполнению операций 328-340 (см. фиг. 13). Локальный сервер 114 репутации может содержать менеджер 170 кэша сервера, менеджер 88 локальной среды и менеджер 168 серверного облака. Работа менеджера 170 кэша сервера и менеджера 168 серверного облака может походить на работу менеджера 70 кэша и, соответственно, менеджера 68 облака, описанных выше в отношении фиг. 7, 8 и фиг. 13. В некоторых вариантах осуществления изобретения, если менеджер 58 репутации клиентской системы 10 требует данных репутаций целевого исполняемого модуля, менеджер 10 может составить запрос 160 репутации, содержащий индикатор идентичности (например, идентификатор хэша) целевого модуля, и посредством локальной сети 120 передать запрос 160 в локальный сервер 114 репутации. На этапе 352 менеджер 170 серверного облака просматривает данные репутации, отыскивая соответствующий модуль в серверном кэше 22 репутации. Если такие данные в кэше 22 существуют, то выполнение переходит к описанному ниже этапу 362. Если данные репутации в кэш не помещены, то на этапе 354 менеджер 88 локальной среды ищет данные репутации в специфической для среды базе 24 данных. Если данные репутации в базе 24 данных найдены, выполнение переходит к этапу 362. Если данные репутации в базе 24 данных не найдены, то на этапе 358 менеджер 168 серверного облака может запросить данные репутации у центрального сервера 14 репутации. Затем на этапе 360 менеджер 168 серверного облака в соответствии с ответом, полученным от сервера 14, определяет, нашел ли запрошенные данные центральный сервер 14 репутации. Если менеджер 168 облака получает данные репутации, например, индикатор 78 репутации облака соответствующего модуля, то локальный сервер 114 репутации рассчитывает индикатор 80 репутации соответствующего модуля (см. например, описание этапов 336-338 на фиг. 13), а на этапе 364 менеджер кэша сервера сохраняет индикатор 80 в серверном кэше 22 репутации. На этапе 362 локальный сервер 114 репутации может передать запрашивающему клиенту 10 ответ на запрос 160 репутации. В некоторых вариантах осуществления изобретения этап 362 может дополнительно содержать расчет индикатора 82 репутации процесса (см., например, описание этапов 344-346 на фиг. 13).

В еще одном примере выполнения этапа 322 (фиг. 12) событие-триггер содержит загрузку целевым процессом исполняемого модуля, например, DLL. В некоторых вариантах осуществления изобретения загрузка нового модуля может изменить индикатор 82 репутации целевого процесса. Например, загрузка ненадежного модуля может понизить репутацию процесса с позитивной или надежной на нейтральную или неизвестную, или даже на негативную или ненадежную. Перерасчет индикатора 82, например, может содержать выполнение этапов 328-340 (фиг. 13), чтобы определить индикатор 80 репутации соответствующего модуля, причем выбранный модуль представляет собой недавно загруженный исполняемый модуль. В этом случае модуль 72 принятия решения может определить индикатор 82 способом, аналогичным способу, описанному выше в отношении этапов 344-346. Например, индикатор 83с оценки процесса может быть определен в соответствии с таблицей решений (см., например, таблицу 1), а значения 83а параметров сканирования процесса могут быть обновлены в результате комбинации текущих значений 83а со значениями 81а параметров сканирования недавно загруженного модуля с применением таких операторов, как объединение/ИЛИ и пересечение/И.

В еще одном примере выполнения этапа 322 (фиг. 12) событие-триггер содержит выгрузку целевым процессом исполняемого модуля, например, DLL. В таких случаях модуль 72 принятия решения может повторно рассчитать индикатор 82 репутации целевого процесса, чтобы отразить новый состав модуля целевого процесса. Чтобы повторно рассчитать индикатор 82, в некоторых вариантах осуществления изобретения могут выполняться последовательности этапов, показанных на фиг. 13. В некоторых вариантах осуществления изобретения модуль 72 принятия решения может решать, рассчитывать ли повторно индикатор 82 репутации процесса в соответствии с текущим индикатором 83с оценки целевого процесса. Например, если целевой процесс в текущий момент оценивается, как негативный или ненадежный, то модуль принятия решения может сохранить оценку целевого процесса, как негативного или ненадежного на время жизни этого процесса, не рассчитывая повторно индикатор 82.

На фиг. 15 показан пример последовательности этапов, выполняемых модулем 72 принятия решения после получения уведомления о событии, связанном с безопасностью (этап 320 на фиг. 12). В некоторых вариантах осуществления изобретения уведомление 64 о событии, связанном с безопасностью (фиг. 7) содержит индикатор подозреваемого процесса и/или подозреваемого модуля, а также индикатор типа события, связанного с безопасностью и инициирующего соответствующее уведомление. Например, уведомление 64 может показывать, что модуль X процесса Y внедрил код в процесс Z. В этом примере X может представлять собой подозреваемый модуль, Y - подозреваемый процесс, а внедрение кода может быть триггером события, связанного с безопасностью. На этапе 372 оценивается уведомление 64 системы безопасности, например, чтобы получить из него тип события, связанного с безопасностью, и идентифицировать подозреваемый процесс и/или подозреваемый модуль. На этапе 374 модуль 72 принятия решения может определить, оправдывает ли событие, связанное с безопасностью и идентифицированное на этапе 372, выдачу команды движку 56 защиты от вредоносных программ, войти в специальный режим работы, называемый чрезмерно строгим режимом. Если выдача такой команды оправдана, то на этапе 376 модуль 72 принятия решения может передать в движок 56 защиты от вредоносных программ предупреждение 66 системы защиты (фиг. 6).

В некоторых вариантах осуществления изобретения активация чрезмерно строгого режима, кроме того, может содержать временное, на определенный период времени, например, на несколько минут, изменение индикаторов 83с оценки всех процессов, выполняющихся в данное время, на негативные или ненадежные. Этот период времени рассматривается, как временной интервал чрезмерно строгого режима. По истечении временного интервала чрезмерно строгого режима индикаторы репутации всех процессов могут быть восстановлены до значений, существовавших до предупреждения 66 системы защиты. В некоторых вариантах осуществления изобретения временной интервал чрезмерно строгого режима зависит от события: некоторые события, связанные с безопасностью, гарантируют активацию чрезмерно строгого режима на более продолжительное время, чем другие события, связанные с безопасностью. Если приблизительно в одно и то же время происходит более чем одно событие, связанное с безопасностью, то некоторые варианты осуществления изобретения устанавливают временной интервал чрезмерно строгого режима в значение, по меньшей мере равное самому большому временному интервалу всех отдельных событий, связанных с безопасностью. Если последовательно происходит несколько событий, связанных с безопасностью, то временной интервал чрезмерно строгого режима может накапливаться, или он может быть определен так, чтобы он истекал одновременно с

временным интервалом чрезмерно строгого режима события, являющегося последним событием в последовательности событий. Более подробно чрезмерно строгий режим описан ниже.

Если введение чрезмерно строгого режима не оправдано, то на этапе 378 модуль 72 принятия решения обновляет индикатор 81с оценки подозреваемого модуля в соответствии с типом события, связанного с безопасностью и определенного на этапе 372. Обновление индикатора оценки модуля, например, может показывать снижение оценки подозреваемого модуля с «надежный» до «нейтральный» или «ненадежный», в зависимости от серьезности события, связанного с безопасностью. На этапе 380 модуль 72 принятия решения может дать менеджеру 70 кэша команду на хранение в кэше 222 обновленных данных репутации подозреваемого модуля. Кроме того, на этапе 380 может включаться передача в систему 18 обновления базы данных репутационного отклика, выдача системе 18 команды на включение обновления в центральную базу данных 16 репутации (подробности см. ниже).

Затем модуль 72 принятия решения может идентифицировать ряд процессов, использующих в данный момент времени экземпляры подозреваемого модуля. Идентификация таких процессов может происходить в соответствии с данными, управляемыми и обеспечиваемыми монитором 74 активности. Затем модуль принятия решения может выполнить в цикле последовательность этапов 384-390 для каждого процесса соответствующего набора процессов, чтобы обновить индикаторы 82 репутации этих процессов. Такое обновление делает возможным эффективную передачу через клиентскую систему 10 новых данных безопасности, относящихся к подозреваемому модулю. На этапе 384 выбирают процесс из набора процессов, в текущий момент времени имеющих экземпляр подозреваемого модуля, загруженный в их пространства памяти. Этап 386 позволяет повторно рассчитать индикатор 82 репутации выбранного процесса, например, применяя последовательность этапов, описанных выше со ссылкой на фиг. 13. В результате выполнения этапа 386 репутация некоторых процессов может быть понижена с «позитивной/надежной» до «нейтральной/неизвестной», или до «негативной/ненадежной». Аналогично оценка некоторых процессов может быть понижена с «нейтральный/неизвестный» до «негативный/ненадежный». На этапе 388 в движок 56 защиты от вредоносных программ может быть передан обновленный индикатор 82 репутации выбранного процесса. На этапе 390 модуль 72 принятия решения проверяет, имеются ли пока еще не обновленные процессы, и если такие процессы существуют, происходит возврат к этапу 384.

На фиг. 16 показан пример последовательности этапов, выполняемых движком 56 защиты от вредоносных программ согласно некоторым вариантам осуществления настоящего изобретения. Движок 56 может выполняться одновременно с менеджером 58 репутации и обмениваться с менеджером 58 репутации уведомлениями и/или данными репутации, как описано выше. В ходе выполнения последовательности этапов 394-396 движок 56 защиты от вредоносных программ конфигурируют так, что он ждет наступления события, в то же время осуществляя операции защиты от вредоносных программ, например, обнаружение и/или удаление вредоносного программного обеспечения. Обнаружение вредоносного программного обеспечения может содержать мониторинг набора процессов и/или исполняемых модулей, выполняющихся в текущий момент в клиентской системе 10 и, например, перехват определенных действий, выполняемых соответствующими процессами и/или модулями, и определение, являются ли эти действия вредоносными или указывающими на вредоносность. В некоторых вариантах осуществления изобретения элементы 76a-76d защиты движка 56 благодаря

настройке параметров сканирования, специфичных для этих элементов, могут быть выполнены независимо друг от друга. Благодаря настройке значения этих параметров сканирования движок 56 может быть сконфигурирован для работы в каждом из нескольких отдельных режимов и/или рабочих протоколов. В некоторых вариантах осуществления изобретения такие параметры сканирования могут быть установлены независимо для каждого процесса и/или исполняемого модуля, так что движок 56 может сканировать или осуществлять мониторинг каждого такого процесса или модуля, применяя потенциально особый протокол. Примерами значений параметров сканирования, специфических для процесса, являются значения 83а параметров сканирования индикатора 82 репутации процесса, передаваемые менеджером 58 репутации в движок 56, как показано выше.

На этапе 396 определяют, произошло ли событие, и если событие не произошло, происходит возврат к этапу 394. Такие события содержат события, связанные с безопасностью и происходящие в клиентской системе 10, а также получение предупреждений 64 системы защиты и индикаторов 82 репутации процесса от менеджера 58 репутации. Если произошло событие, то на этапе 398 движок 56 может определить, содержит ли это событие прием предупреждения системы защиты, и если нет, движок 56 может перейти к этапу 400. В некоторых вариантах осуществления изобретения предупреждения 66 системы защиты дают движку 56 команду войти в чрезмерно строгий режим или выйти из этого режима. Чтобы войти в чрезмерно строгий режим, на этапе 404 движок 56 может сбросить параметры сканирования элементов 76а-76d защиты и установить заданный набор значений, соответствующий строгому протоколу системы защиты, таким образом, аннулируя любые специфические для процесса значения, полученные от менеджера 58 репутации. В некоторых вариантах осуществления изобретения строгий протокол системы защиты, активируемый в чрезмерно строгом режиме, содержит активацию элементов защиты, которые в остальных случаях могут не использоваться, чтобы сберечь вычислительные ресурсы. Эти элементы могут содержать программы и/или процедуры обнаружения, специально направленные на тип угрозы, вызвавшей соответствующее предупреждение системы безопасности. Кроме того, чрезмерно строгий режим, помимо прочего, может содержать активацию дополнительного набора методов обнаружения посредством эвристического анализа и блокировку набора оптимизаций, применяемую для ускорения операций по защите от вредоносных программ. В некоторых вариантах осуществления изобретения чрезмерно строгий режим может быть активирован на ограниченный период времени, указываемый в предупреждении 66 системы безопасности. Срок чрезмерно строгого режима может истекать автоматически по прошествии указанного периода, или он может быть выключен в ответ на получение извещения 66 системы защиты, в явной форме дающего движку 56 команду на выход из чрезмерно строгого режима. В некоторых вариантах осуществления изобретения истечение срока чрезмерно строгого режима содержит сброс параметров сканирования и установку значений, специфических для данного процесса и/или модуля и зарегистрированных перед активацией чрезмерно строгого режима.

На этапе 400 движок 56 защиты от вредоносных программ определяет, содержит ли событие получение от менеджера 58 репутации индикатора репутации процесса, и если нет, то движок 56 переходит к этапу 402. Если индикатор репутации был принят, то на этапе 406 движок защиты от вредоносных программ может обновить значения параметров сканирования и установить новые значения, показываемые принятым индикатором репутации.

На этапе 402 движок 56 определяет, является ли это событие событием, связанным с безопасностью (например, один процесс внедрил библиотеку в другой процесс), и если нет, движок 56 возвращается к этапу 394. Если событие, связанное с безопасностью, действительно было обнаружено, на этапе 408 идентифицируется подозреваемый модуль и/или подозреваемый процесс, вызывающий соответствующее событие, связанное с безопасностью, и на этапе 410 движок 56 передает в менеджер 58 репутации уведомление 64 о событии, связанном с безопасностью, показывающее идентификацию подозреваемого модуля и тип события, связанного с безопасностью.

На фиг. 17 показан пример конфигурации системы 18 обновления репутации (см., например, фиг. 1) согласно некоторым вариантам осуществления настоящего изобретения. Система 18 может быть сконфигурирована для поддержания в актуальном состоянии данных репутации, сохраненных в центральной базе данных 16, благодаря изменению текущих данных репутации известных модулей в ответ на получение от клиентской системы 10 репутационного отклика 90, относящегося к соответствующим модулям, и/или благодаря добавлению в центральную базу данных 16 репутации данных репутации, определенных для неизвестных модулей, идентифицированных в репутационном отклике 90. В некоторых вариантах осуществления изобретения известные модули представляют собой программные объекты, для которых, если отклик 90 получен, в базе данных 16 данные репутации уже имеются, в то время как неизвестные модули представляют собой объекты, для которых данные репутации в базе данных 16 не имеются. Информация по неизвестным модулям может быть получена от клиентских систем 10a-10e и/или от поставщиков программного обеспечения, выпускающих новые продукты или обновления продуктов.

В некоторых вариантах осуществления изобретения система 18 обновления репутации содержит анализатор 92 отклика, движок 94 анализа модуля, соединенный с анализатором 92 отклика, и модуль 96 обновления базы данных, соединенный с анализатором 92 и движком 94. Движок 94 анализа модуля может быть сконфигурирован для оценки репутации неизвестных модулей и составления индикаторов 78 репутации облака для этих модулей. В некоторых вариантах осуществления изобретения движок 94 сконфигурирован для получения записи действий и/или поведений каждого модуля, благодаря которой оператор и/или машина может вывести оптимальные значения параметров сканирования для соответствующего неизвестного модуля. Такой вывод и/или оптимизацию значений параметров сканирования могут выполнить известным в области техники способом. Чтобы получить эти записи, движок 94 может применить компоненты обнаружения события, которые могут быть сконфигурированы для перехвата событий при исполнении программы, например, запуска порождаемого объекта или внедрения кода соответствующим неизвестным модулем. В некоторых вариантах осуществления изобретения движок 94 анализа модуля может быть дополнительно сконфигурирован для определения значений 79a параметров сканирования облака и/или индикатора 79b истечения срока кэша для соответствующего неизвестного модуля. Такие определения могут быть полностью автоматическими определениями, осуществляемыми компонентами движка 94, или они могут контролироваться людьми-операторами.

На фиг. 18 показан пример последовательности этапов, выполняемых системой 18 обновления при получении репутационного отклика 90 от клиентской системы 10. В некоторых вариантах осуществления изобретения репутационный отклик 90 содержит индикатор идентичности (например, идентификатор хэша) исполняемого модуля, показывающий соответствующий модуль, как предмет отклика 90. Если подозреваемый

модуль известен, то репутационный отклик 90 может дополнительно содержать индикатор изменения репутации соответствующего модуля. В некоторых вариантах осуществления изобретения эта репутация описывается индикатором оценки модуля (см., например, позицию 81с на фиг. 10-В), показывающим, например, что

5 соответствующий модуль имеет позитивную, нейтральную или негативную репутацию. Изменения репутации могут быть инициированы соответствующим модулем, вызывающим в клиентской системе 10 событие, связанное с безопасностью (см. также выше описание, относящееся к этапам 402, 408 и 410 на фиг. 16). Например, если модуль производит действие, указывающее на вредоносность, то его репутация может быть

10 понижена с позитивной/надежной до нейтральной/ неизвестной или даже негативной/ ненадежной. В таких случаях репутационный отклик 90 может дополнительно содержать индикатор, показывающий тип события, связанного с безопасностью (например, внедрение кода) и вызванного соответствующим модулем. Если репутационный отклик 90 относится к неизвестному модулю, то отклик 90 может содержать участок кода,

15 например, весь исполняемый код соответствующего модуля и/или метаданные, например, имя файла модуля, размер в байтах, метку времени и путь к соответствующему модулю.

В ходе последовательности этапов 412-414 анализатор 92 отклика определяет тип отклика (например, обновление репутации), показываемый репутационным откликом 90, и определяет, к какому исполняемому модулю - известному или неизвестному -

20 относится отклик 90. Если соответствующий модуль известен, то на этапе 416 система 18 применяет модуль 96 обновления базы данных, чтобы обновить данные репутации, имеющиеся в настоящее время в центральной базе данных 16 репутации и относящиеся к соответствующему модулю, чтобы отразить изменение репутации, показываемое откликом 90. В некоторых вариантах осуществления изобретения, если репутация

25 соответствующего модуля понижена с позитивной/надежной до нейтральной/неизвестной или до негативной/ненадежной, то модуль 96 обновления базы данных может удалить из центральной базы 16 данных репутации запись о репутации соответствующего модуля. В таких вариантах осуществления изобретения база 16 данных может эффективно хранить только записи исполняемых модулей, оцениваемых, как позитивные/надежные.

Если отклик 90 показывает неизвестный модуль, то на этапе 418 система 18

30 обновления репутации передает данные соответствующего модуля для анализа в движок 94 анализа модуля. На этапе 418 благодаря анализу участка кода соответствующего модуля движок 94 может определить, является ли соответствующий модуль, вероятно, вредоносным. Кроме того, движок 94 может определить ряд свойств неизвестного

35 модуля, например, определить, осуществляет ли подозреваемый модуль любое из заранее определенных действий (например, загружает файлы из сети, записывает файлы на диск, внедряет код и так далее). В некоторых вариантах осуществления изобретения этап 418 также может содержать классификацию неизвестного модуля в один из нескольких классов или категорий исполняемых модулей (например, главный

40 исполняемый файл, общая библиотека и так далее).

На этапе 420 движок 94 анализа модуля может определить индикатор репутации облака соответствующего модуля, включая набор значений параметров сканирования, применяемых для конфигурации движков 56 для защиты от вредоносных программ. Затем на этапе 422 система 18 обновления репутации может применить модуль

45 обновления базы данных, чтобы сохранить в центральной базе 16 данных индикатор 78 репутации облака, определенный для неизвестного модуля. В некоторых вариантах осуществления изобретения этап 422 осуществляется только в отношении неизвестных модулей, получивших от движка 94 позитивную или надежную оценку репутации.

Системы и способы, описанные выше в качестве примера, позволяют защитить клиентскую систему, например, компьютерную систему, от вредоносного программного обеспечения, например, от вирусов, троянов и шпионского программного обеспечения. В некоторых вариантах осуществления изобретения менеджер репутации выполняется
5 одновременно с движком защиты от вредоносных программ. Движок защиты от вредоносных программ осуществляет такие операции, как обнаружение вредоносных программ, выполняющихся в соответствующей клиентской системе, и/или удаление, или блокировку таких программ. Для каждого процесса, выполняющегося в клиентской системе, менеджер репутации может передать в движок защиты от вредоносных
10 программ индикатор репутации, показывающий степень уверенности в том, что соответствующий процесс не вредоносен.

В обычных системах защиты от вредоносных программ программные объекты сканируются независимо от их репутации. Напротив, в некоторых вариантах осуществления настоящего изобретения движок защиты от вредоносных программ
15 может обеспечить для надежных объектов преференциальный режим. Например, движок защиты от вредоносных программ может использовать для сканирования надежных объектов протокол, требующий меньших вычислительных затрат, чем при сканировании неизвестных объектов. В одном таком примере при сканировании надежных объектов подбор параметров движка защиты от вредоносных программ может быть
20 заблокирован. Благодаря уменьшению затрат вычислительных ресурсов, связанных со сканированием надежных объектов, такой подход позволяет значительно усовершенствовать эффективность защиты от вредоносных программ.

Благодаря настройке ряда параметров сканирования в некоторых вариантах осуществления изобретения движок защиты от вредоносных программ может быть
25 выполнен с возможностью работы в нескольких разных режимах. Благодаря настройке значений параметров сканирования в зависимости от специфики процесса движок защиты от вредоносных программ может сканировать каждый процесс, применяя специфический для этого процесса способ или протокол. В некоторых вариантах осуществления изобретения значения параметров сканирования выбирают в
30 соответствии с репутацией соответствующего объекта, следовательно, способ или протокол защиты от вредоносных программ может изменяться в соответствии с репутацией сканируемого объекта.

В случае некоторых обычных систем защиты от вредоносных программ делают попытки уменьшить затраты вычислительных ресурсов на сканирование, применяя
35 ряд оптимизаций программ защиты от вредоносных программ. Такая оптимизация может содержать определение в соответствии с рядом свойств соответствующего объекта, относится ли целевой объект к особой категории объектов, и предоставление преференциальной обработки объектам, относящимся к соответствующей категории. Такие оптимизации не всегда могут быть эффективными при переходе от одного типа
40 вредоносных программ к другому типу, и они могут использоваться вредоносными объектами, специально предназначенными для уклонения от обнаружения посредством включения в соответствующую категорию. Напротив, в некоторых вариантах осуществления настоящего изобретения преференциальная обработка представляется объекту не в соответствии со свойством соответствующего объекта, а в соответствии
45 с репутацией, показывающей степень уверенности в том, что соответствующий объект не вредоносен. Такой подход позволяет без труда переходить от одного типа или категории вредоносных программ к другому типу, включая возникающие угрозы. Кроме того, менеджер репутации работает независимо от движка защиты от

вредоносных программ, который может быть усовершенствован так, чтобы ввести новые способы и процедуры сканирования, не оказывая влияния на работу менеджера репутации.

В некоторых вариантах осуществления настоящего изобретения репутация целевого программного объекта может быть определена в соответствии с репутацией набора стандартных блоков соответствующего процесса. Примеры таких стандартных блоков, помимо прочего, содержат главный исполняемый файл процесса и общую библиотеку, загружаемую соответствующим процессом. Репутация каждого стандартного блока может быть статичной, индикатор такой репутации может быть сохранен в локальном репозитории (например, в локальном кэше) и/или в удаленной репозитории (например, в центральной базе данных репутации) и повторно использоваться для каждого экземпляра соответствующего стандартного блока. Напротив, репутация самого целевого объекта может быть динамически изменяемой, то есть рассчитываться повторно менеджером репутации в ответ на события, связанные с безопасностью, и/или события в жизненном цикле целевого объекта.

Репутация стандартного блока, например, общей библиотеки, может показывать степень уверенности в том, что соответствующий блок не является вредоносным. Такие репутации отдельных стандартных блоков могут быть определены в соответствии с поведением множества отдельных экземпляров этих стандартных блоков, распределенным по нескольким клиентским системам, имеющих удаленное подключение к центральному серверу репутации. В некоторых вариантах осуществления настоящего изобретения сохраненные данные репутации стандартного блока могут быть обновлены в ответ на событие, связанное с безопасностью и затрагивающее экземпляр соответствующего стандартного блока. В одном таком примере, если общая библиотека производит действие, указывающее на вредоносность, то репутация соответствующей библиотеки может быть понижена с «надежной» до «неизвестной» или «ненадежной». Обновленный индикатор репутации может быть сохранен в локальном кэше и/или передан в центральную базу данных репутации. Благодаря таким конфигурациям любые изменения в репутации могут быстро распространить на другие локальные процессы, применяя соответствующую общую библиотеку, и, кроме того, на другие клиентские системы, соединенные с сервером репутации.

В некоторых вариантах осуществления изобретения сохраненные данные репутации стандартных блоков содержат набор значений параметров сканирования, применяемых для конфигурирования движка защиты от вредоносных программ, чтобы осуществлять мониторинг соответствующего стандартного блока и/или процессов, содержащих экземпляр соответствующего стандартного блока. Такие значения параметров, например, показывающие, какой метод эвристического анализа необходимо использовать для мониторинга поведения соответствующего объекта, могут быть определены людьми-операторами или в автоматическом режиме машиной. Оптимальные значения параметров сканирования могут быть определены, например, благодаря мониторингу выполнения соответствующего исполняемого модуля в контролируемой среде и определению набора свойств поведения и/или статичных свойств модуля. Значения параметров сканирования могут быть обновлены, чтобы эффективно обращаться с недавно обнаруженными типами и вариантами вредоносных объектов. Благодаря вышеописанному механизму действия сервера и/или просмотра данных в кэше затем такие обновления могут быть распространены на все клиентские системы, соединенные с сервером репутации.

Специалисту в области техники очевидно, что описанные выше варианты

осуществления изобретения могут быть изменены различным образом без выхода за пределы объема правовой охраны настоящего изобретения. Соответственно, объем правовой охраны настоящего изобретения должен определяться прилагаемой формулой изобретения и ее эквивалентами, установленными законом.

5

(57) Формула изобретения

1. Клиентская система, содержащая:

- память; и

10 - по меньшей мере один аппаратный процессор, соединенный с памятью и сконфигурированный для выполнения движка защиты от вредоносных программ, сконфигурированного для мониторинга целевого процесса на наличие вредоносной активности, причем целевой процесс выполняется на клиентской системе, причем целевой процесс содержит экземпляр главного исполняемого модуля и экземпляр общей библиотеки, при этом по меньшей мере один аппаратный процессор дополнительно

15

сконфигурирован для:

- приема от сервера первого индикатора репутации модуля главного исполняемого модуля и второго индикатора репутации модуля общей библиотеки, причем первый индикатор репутации модуля определен в соответствии с поведением еще одного экземпляра главного исполняемого модуля, при этом сервер сконфигурирован для

20

- выполнения транзакций по защите от вредоносных программ с множеством клиентских систем, содержащим указанную клиентскую систему, причем первый индикатор репутации модуля содержит индикатор первого набора правил мониторинга и второй индикатор репутации модуля содержит индикатор второго набора правил мониторинга;

25

- определения, в ответ на прием первого и второго индикатора репутации модуля, является ли целевой процесс, вероятно, вредоносным, в соответствии с первым и вторым индикатором репутации модуля; и

- в ответ на определение, является ли целевой процесс, вероятно, вредоносным, и если целевой процесс, вероятно, не является вредоносным,

30

- комбинирования первого и второго набора правил мониторинга в комбинированный набор правил мониторинга; и

- конфигурирования движка защиты от вредоносных программ для мониторинга целевого процесса в соответствии с комбинированным набором правил мониторинга.

35

2. Клиентская система по п. 1, в которой по меньшей мере один аппаратный процессор сконфигурирован для определения комбинированного набора правил мониторинга в соответствии с объединением первого и второго наборов правил мониторинга.

3. Клиентская система по п. 1, в которой по меньшей мере один аппаратный процессор сконфигурирован для определения комбинированного набора правил мониторинга в соответствии с пересечением первого и второго наборов правил мониторинга.

40

4. Клиентская система по п. 1, в которой выбранное правило из первого набора правил мониторинга включает выбранный метод эвристического анализа, применяемый для определения, является ли целевой процесс вредоносным.

5. Клиентская система по п. 1, в которой выбранное правило из первого набора правил мониторинга обеспечивает выдачу команды движку защиты от вредоносных программ на определение, выполняет ли целевой процесс выбранное действие.

45

6. Клиентская система по п. 1, в которой обеспечена возможность выполнения еще одного экземпляра главного исполняемого модуля во второй клиентской системе из множества клиентских систем.

7. Клиентская система по п. 1, в которой по меньшей мере один аппаратный процессор

дополнительно сконфигурирован, в ответ на конфигурирование движка защиты от вредоносных программ, для:

- обнаружения действия, осуществляемого целевым процессом, причем действие содержит загрузку другого исполняемого модуля в целевой процесс, причем другой исполняемый модуль является отдельным от главного исполняемого модуля и общей библиотеки;

- повторной оценки, в ответ на обнаружение указанного действия, целевого процесса для определения, является ли целевой процесс, вероятно, вредоносным, в соответствии с третьим индикатором репутации модуля другого исполняемого модуля, причем третий индикатор репутации модуля принят от сервера; и

- реконфигурирования, в ответ, движка защиты от вредоносных программ в соответствии с третьим набором правил мониторинга, показанным третьим индикатором репутации модуля.

8. Клиентская система по п. 1, в которой определение, является ли целевой процесс, вероятно, вредоносным, содержит:

- идентификацию всех исполняемых модулей, загружаемых целевым процессом;

- определение, является ли каждый модуль из всех исполняемых модулей, вероятно, вредоносным, в соответствии с индикатором репутации модуля, определенным для экземпляра каждого модуля; и

- определение, в ответ, если каждый модуль из всех исполняемых модулей, вероятно, не является вредоносным, что целевой процесс, вероятно, не является вредоносным.

9. Клиентская система по п. 8, в которой определение, является ли целевой процесс, вероятно, вредоносным, дополнительно содержит, в ответ на определение того, является ли каждый модуль, вероятно, вредоносным, и если по меньшей мере один модуль из всех исполняемых модулей, вероятно, является вредоносным, определение, что целевой процесс, вероятно, является вредоносным.

10. Клиентская система по п. 1, в которой по меньшей мере один аппаратный процессор сконфигурирован, в ответ на конфигурирование движка защиты от вредоносных программ, для:

- определения, выполняет ли целевой процесс действие, указывающее на вредоносность; и

если целевой процесс выполняет действие, указывающее на вредоносность,

- идентификации подозреваемого модуля целевого процесса, причем подозреваемый модуль выполняет действие, указывающее на вредоносность;

- обновления индикатора репутации модуля для подозреваемого модуля, чтобы показать, что подозреваемый модуль, вероятно, является вредоносным; и

- реконфигурирования, в ответ на обновление индикатора репутации модуля для подозреваемого модуля, движка защиты от вредоносных программ для мониторинга целевого процесса в соответствии со строгим протоколом системы защиты, причем строгий протокол системы защиты требует большего количества вычислительных ресурсов, чем комбинированный набор правил мониторинга.

11. Клиентская система по п. 10, в которой по меньшей мере один аппаратный процессор дополнительно сконфигурирован, в ответ на обновление индикатора репутации модуля для подозреваемого модуля, для передачи на сервер обновленного индикатора репутации модуля для подозреваемого модуля.

12. Клиентская система по п. 10, в которой по меньшей мере один аппаратный процессор дополнительно сконфигурирован, в ответ на обновление индикатора репутации модуля для подозреваемого модуля, для:

- идентификации второго процесса, выполняющегося в клиентской системе, причем второй процесс содержит экземпляр подозреваемого модуля; и

- конфигурирования, в ответ на идентификацию второго процесса, движка защиты от вредоносных программ для мониторинга второго процесса в соответствии со строгим протоколом системы защиты.

13. Клиентская система по п. 1, в которой по меньшей мере один аппаратный процессор дополнительно сконфигурирован, в ответ на конфигурирование движка защиты от вредоносных программ, для:

- определения, выполняет ли целевой процесс действие, указывающее на вредоносность; и

- если целевой процесс выполняет действие, указывающее на вредоносность,

- конфигурирования движка защиты от вредоносных программ для применения строгого протокола системы защиты для мониторинга второго процесса, выполняющегося в клиентской системе, независимо от того, является ли второй процесс, вероятно, вредоносным.

14. Клиентская система по п. 1, в которой по меньшей мере один аппаратный процессор дополнительно сконфигурирован, в ответ на определение, является ли целевой процесс, вероятно, вредоносным, и если целевой процесс, вероятно, является вредоносным, для конфигурирования движка защиты от вредоносных программ для мониторинга целевого процесса в соответствии со строгим протоколом системы защиты, причем строгий протокол системы защиты требует большего количества вычислительных ресурсов, чем комбинированный набор правил мониторинга.

15. Сервер, содержащий:

- память; и

- по меньшей мере один аппаратный процессор, соединенный с памятью и сконфигурированный для:

- определения первого индикатора репутации модуля для главного исполняемого модуля и второго индикатора репутации модуля общей библиотеки, причем первый индикатор репутации модуля определен в соответствии с поведением экземпляра главного исполняемого модуля, причем первый индикатор репутации модуля содержит индикатор первого набора правил мониторинга и второй индикатор репутации модуля содержит индикатор второго набора правил мониторинга; и

- передачи, в ответ на определение первого и второго индикатора репутации модуля, первого и второго индикатора репутации модуля в одну клиентскую систему из множества клиентских систем, сконфигурированную для осуществления с сервером транзакций по защите от вредоносных программ,

причем указанная клиентская система сконфигурирована для выполнения движка защиты от вредоносных программ, сконфигурированного для мониторинга целевого процесса на наличие вредоносной активности, при этом целевой процесс содержит еще один экземпляр главного исполняемого модуля и экземпляр общей библиотеки, причем указанная клиентская система дополнительно сконфигурирована для:

- определения, является ли целевой процесс, вероятно, вредоносным, в соответствии с первым и вторым индикатором репутации модуля;

- в ответ на определение, является ли целевой процесс, вероятно, вредоносным, и если целевой процесс, вероятно, не является вредоносным,

- комбинирования первого и второго набора правил мониторинга в комбинированный набор правил мониторинга; и

- конфигурирования движка защиты от вредоносных программ для мониторинга

целевого процесса в соответствии с комбинированным набором правил мониторинга.

16. Сервер по п. 15, в котором клиентская система сконфигурирована для определения комбинированного набора правил мониторинга в соответствии с объединением первого и второго наборов правил мониторинга.

5 17. Сервер по п. 15, в котором клиентская система сконфигурирована для определения комбинированного набора правил мониторинга в соответствии с пересечением первого и второго наборов правил мониторинга.

18. Сервер по п. 15, в котором выбранное правило из первого набора правил мониторинга включает выбранный метод эвристического анализа, применяемый для
10 определения, является ли целевой процесс вредоносным.

19. Сервер по п. 15, в котором выбранное правило из первого набора правил мониторинга обеспечивает выдачу команды клиентской системе на определение, выполняет ли целевой процесс выбранное действие.

20. Сервер по п. 15, в котором обеспечена возможность выполнения экземпляра
15 главного исполняемого модуля во второй клиентской системе из множества клиентских систем.

21. Сервер по п. 15, в котором клиентская система дополнительно сконфигурирована, в ответ на конфигурирование движка защиты от вредоносных программ, для:

- обнаружения действия, осуществляемого целевым процессом, причем действие
20 содержит загрузку другого исполняемого модуля в целевой процесс, причем другой исполняемый модуль является отдельным от главного исполняемого модуля и общей библиотеки;

- повторной оценки, в ответ на обнаружение указанного действия, целевого процесса для определения, является ли целевой процесс, вероятно, вредоносным, в соответствии
25 с третьим индикатором репутации модуля другого исполняемого модуля, причем третий индикатор репутации модуля принят от сервера; и

- реконфигурирования, в ответ, движка защиты от вредоносных программ в соответствии с третьим набором правил мониторинга, показанным третьим индикатором репутации модуля.

30 22. Сервер по п. 15, в котором определение, является ли целевой процесс, вероятно, вредоносным, содержит:

- идентификацию всех исполняемых модулей, загружаемых целевым процессом;
- определение, является ли каждый модуль из всех исполняемых модулей, вероятно, вредоносным, в соответствии с индикатором репутации модуля, определенным для
35 экземпляра каждого модуля; и

- определение, в ответ, если каждый модуль из всех исполняемых модулей, вероятно, не является вредоносным, что целевой процесс, вероятно, не является вредоносным.

23. Сервер по п. 22, в котором определение, является ли целевой процесс, вероятно, вредоносным, дополнительно содержит, в ответ на определение того, является ли
40 каждый модуль, вероятно, вредоносным, и если по меньшей мере один модуль из всех исполняемых модулей, вероятно, является вредоносным, определение, что целевой процесс, вероятно, является вредоносным.

24. Сервер по п. 15, в котором клиентская система дополнительно сконфигурирована, в ответ на конфигурирование движка защиты от вредоносных программ, для:

45 - определения, выполняет ли целевой процесс действие, указывающее на вредоносность; и

если целевой процесс выполняет действие, указывающее на вредоносность,

- идентификации подозреваемого модуля целевого процесса, причем подозреваемый

модуль выполняет действие, указывающее на вредоносность;

- обновления индикатора репутации модуля для подозреваемого модуля, чтобы показать, что подозреваемый модуль, вероятно, является вредоносным; и

- реконфигурирования, в ответ на обновление индикатора репутации модуля для подозреваемого модуля, движка защиты от вредоносных программ для мониторинга целевого процесса в соответствии со строгим протоколом системы защиты, причем строгий протокол системы защиты требует большего количества вычислительных ресурсов, чем комбинированный набор правил мониторинга.

25. Сервер по п. 24, дополнительно сконфигурированный для приема обновленного индикатора репутации модуля для подозреваемого модуля от клиентской системы.

26. Сервер по п. 24, в котором клиентская система дополнительно сконфигурирована, в ответ на обновление индикатора репутации модуля для подозреваемого модуля, для:

- идентификации второго процесса, выполняющегося в клиентской системе, причем второй процесс содержит экземпляр подозреваемого модуля; и

- конфигурирования, в ответ на идентификацию второго процесса, движка защиты от вредоносных программ для мониторинга второго процесса в соответствии со строгим протоколом системы защиты.

27. Сервер по п. 15, в котором клиентская система дополнительно сконфигурирована, в ответ на конфигурирование движка защиты от вредоносных программ, для:

- определения, выполняет ли целевой процесс действие, указывающее на вредоносность; и

- если целевой процесс выполняет действие, указывающее на вредоносность,

- конфигурирования движка защиты от вредоносных программ для применения строгого протокола системы защиты для мониторинга второго процесса,

- выполняющегося в клиентской системе, независимо от того, является ли второй процесс, вероятно, вредоносным.

28. Сервер по п. 15, в котором клиентская система дополнительно сконфигурирована, в ответ на определение, является ли целевой процесс, вероятно, вредоносным, и если целевой процесс, вероятно, является вредоносным, для конфигурирования движка защиты от вредоносных программ для мониторинга целевого процесса в соответствии со строгим протоколом системы защиты, причем строгий протокол системы защиты требует большего количества вычислительных ресурсов, чем комбинированный набор правил мониторинга.

29. Постоянный машиночитаемый носитель, хранящий команды, которые в случае их выполнения конфигурируют по меньшей мере один процессор клиентской системы, выполняющий целевой процесс, причем целевой процесс содержит экземпляр главного исполняемого модуля и экземпляр общей библиотеки, для:

- приема от сервера первого индикатора репутации модуля для главного исполняемого модуля и второго индикатора репутации модуля общей библиотеки, причем первый

- индикатор репутации модуля определен в соответствии с поведением еще одного экземпляра главного исполняемого модуля, причем сервер сконфигурирован для

- выполнения транзакций по защите от вредоносных программ с множеством клиентских систем, содержащим указанную клиентскую систему, причем первый индикатор репутации модуля содержит индикатор первого набора правил мониторинга и второй

- индикатор репутации модуля содержит индикатор второго набора правил мониторинга;

- определения, в ответ на прием первого и второго индикатора репутации модуля, является ли целевой процесс, вероятно, вредоносным, в соответствии с первым и вторым индикатором репутации модуля; и

в ответ на определение, является ли целевой процесс, вероятно, вредоносным, и если целевой процесс, вероятно, не является вредоносным,

- комбинирования первого и второго набора правил мониторинга в комбинированный набор правил мониторинга; и

5 - конфигурирования движка защиты от вредоносных программ для мониторинга целевого процесса в соответствии с комбинированным набором правил мониторинга.

10

15

20

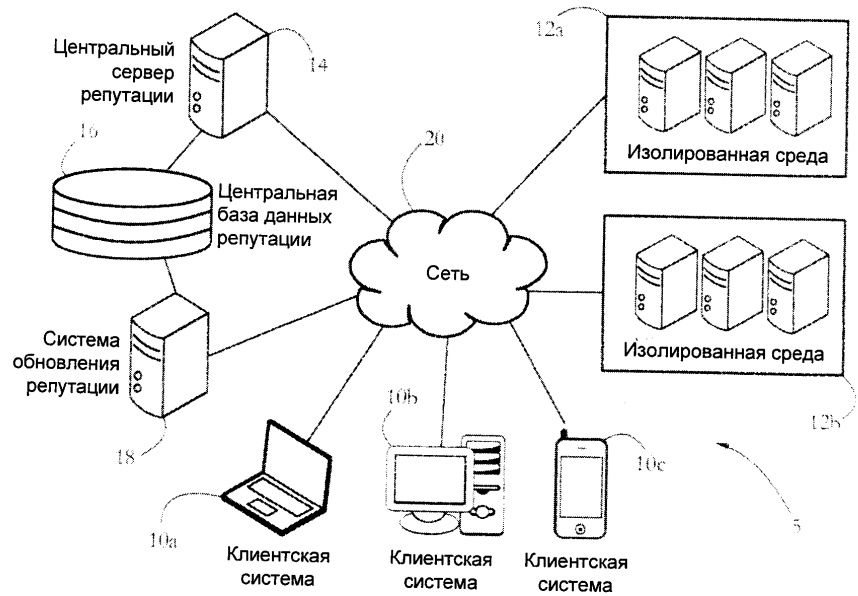
25

30

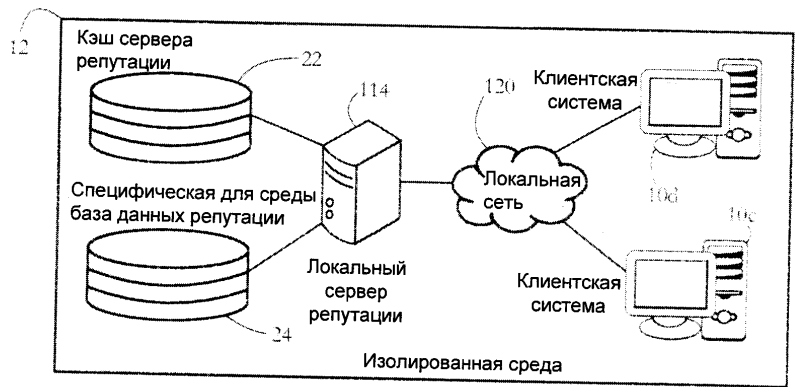
35

40

45



Фиг. 1



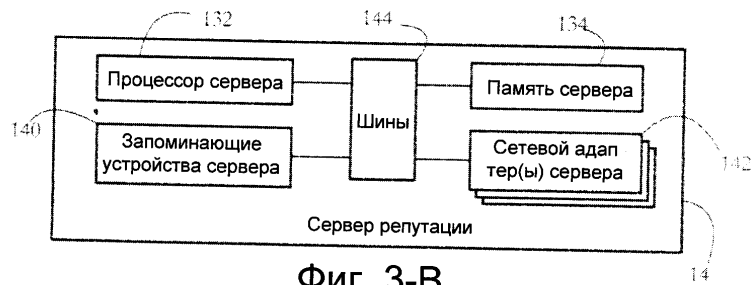
Фиг. 2

36586

2/12



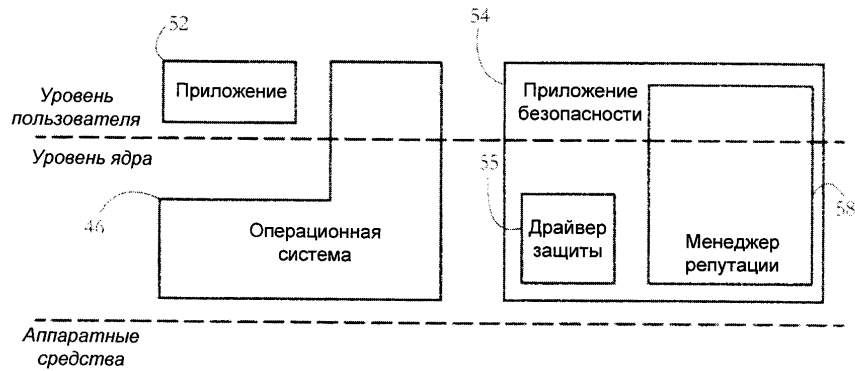
Фиг. 3-А



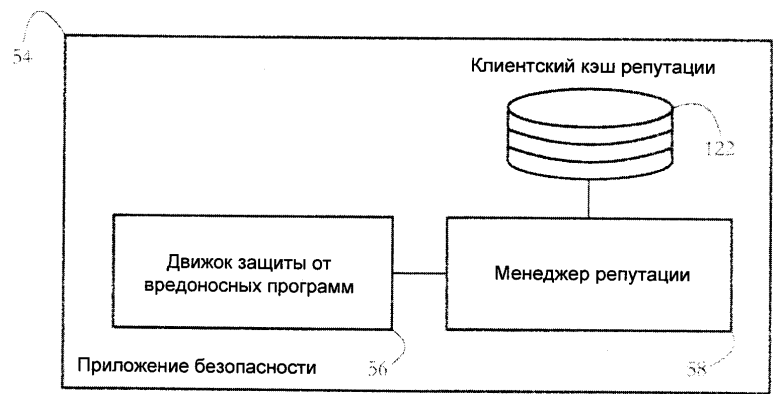
Фиг. 3-В



Фиг. 4



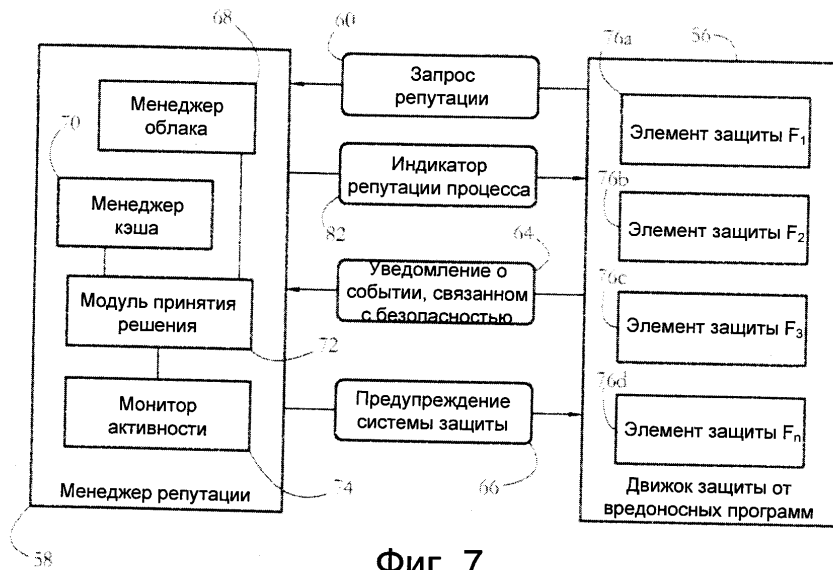
Фиг. 5



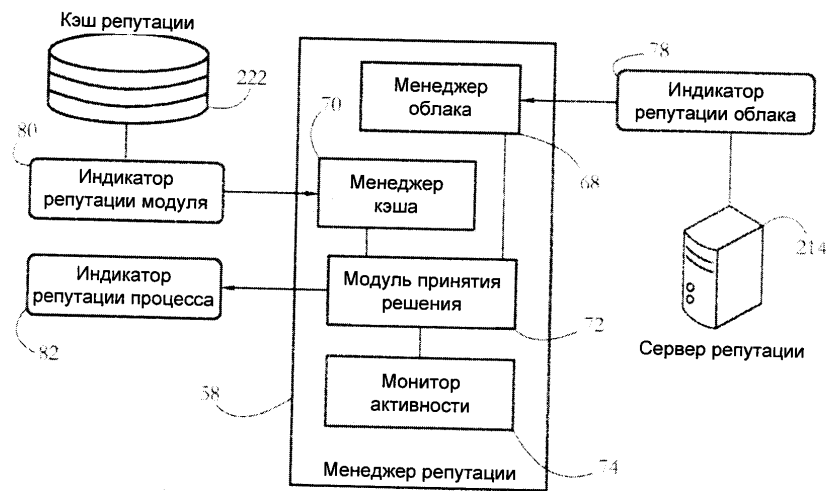
Фиг. 6

36586

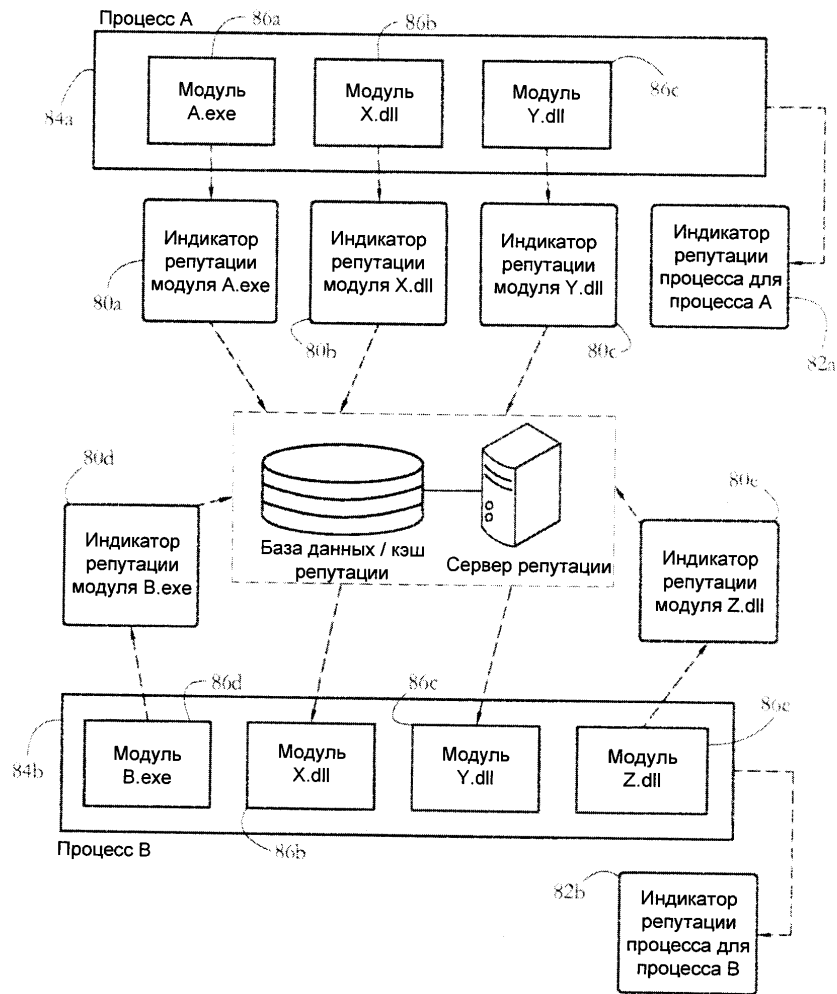
4/12



Фиг. 7



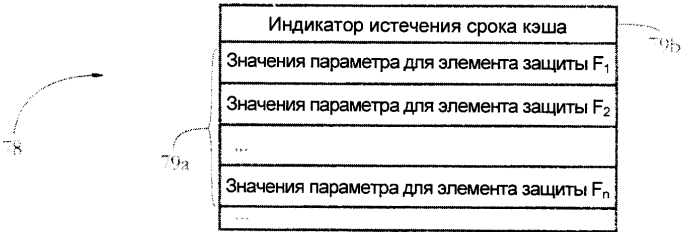
Фиг. 8



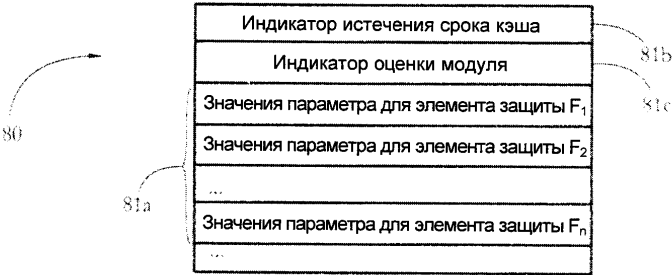
Фиг. 9

36586

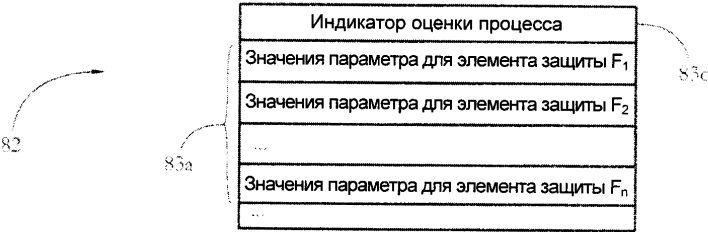
6/12



Фиг. 10А



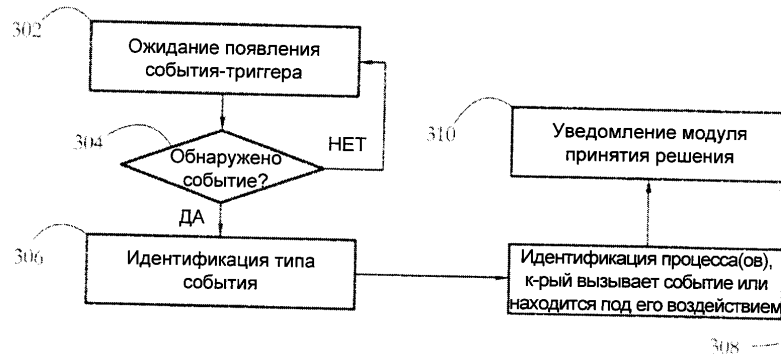
Фиг. 10В



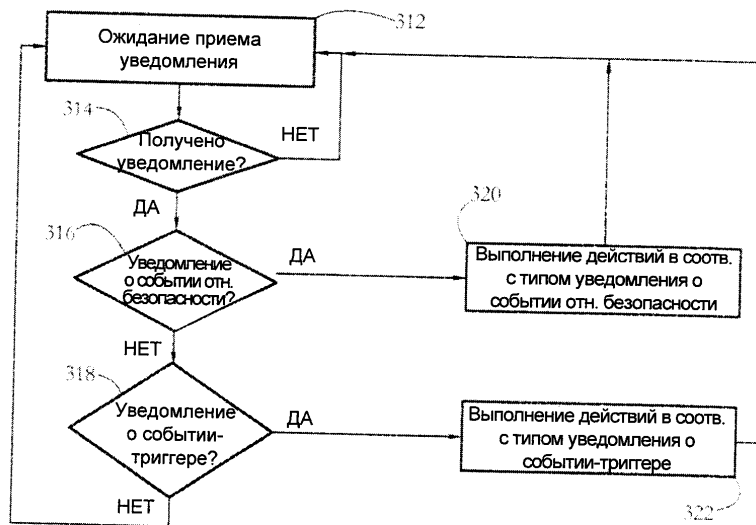
Фиг. 10С

36586

7/12



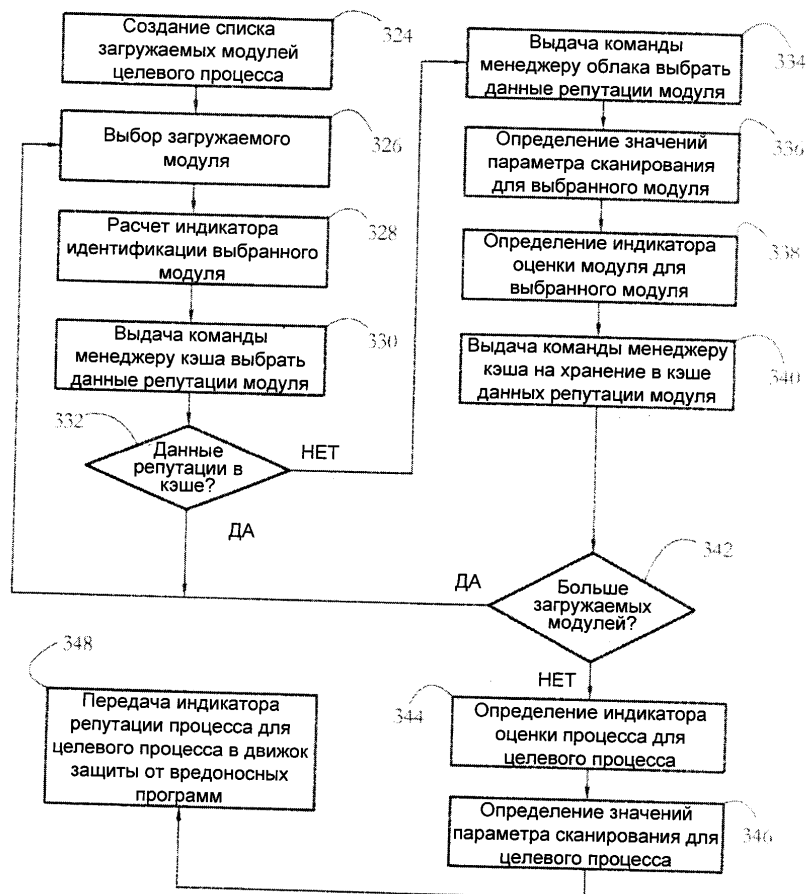
Фиг. 11



Фиг. 12

36586

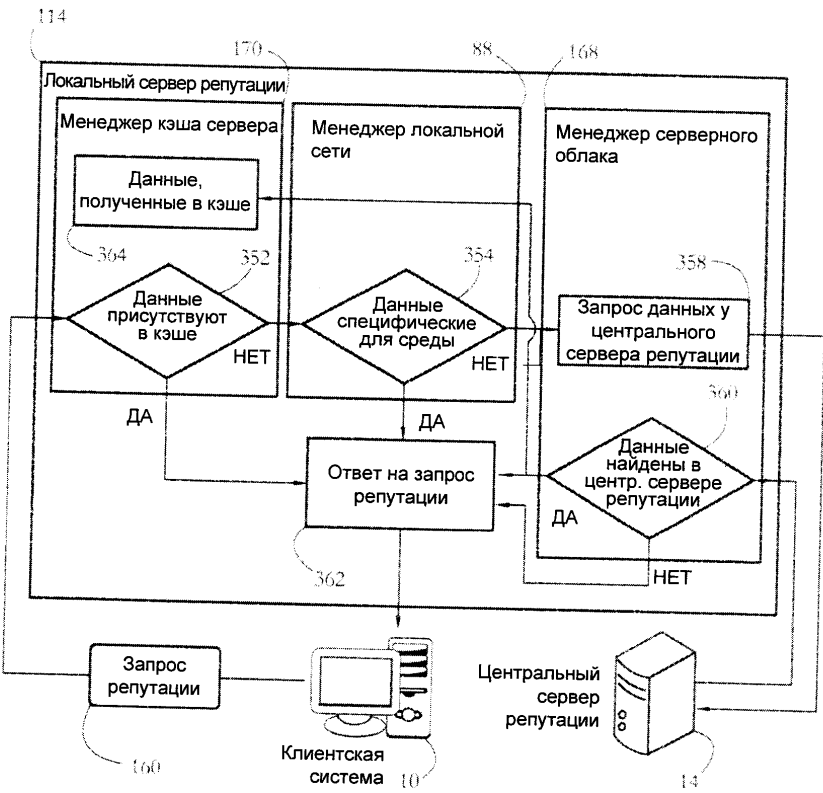
8/12



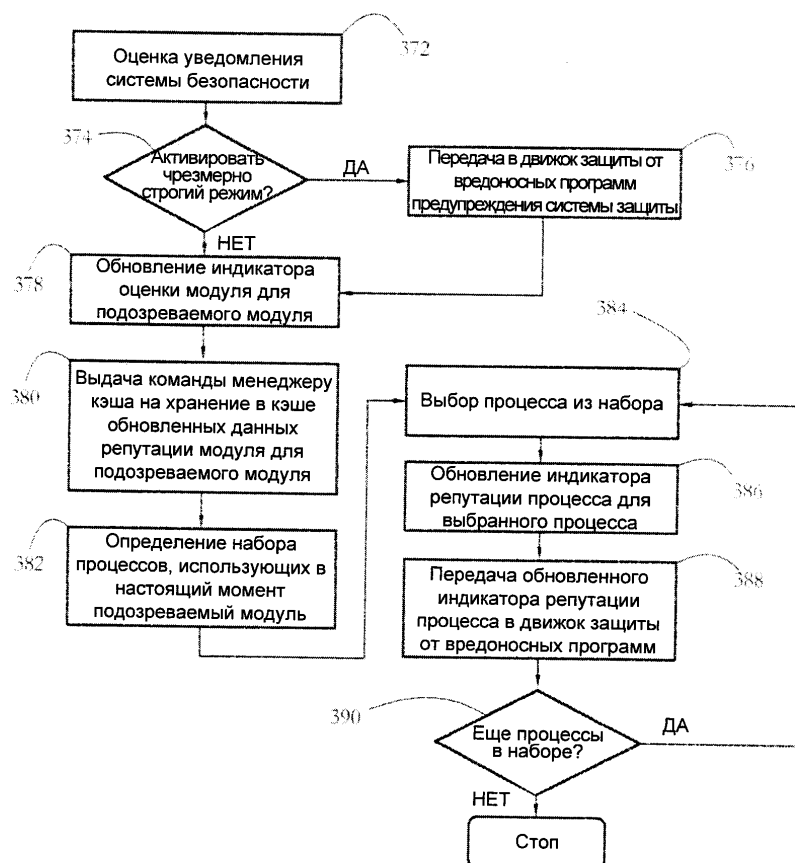
Фиг. 13

36586

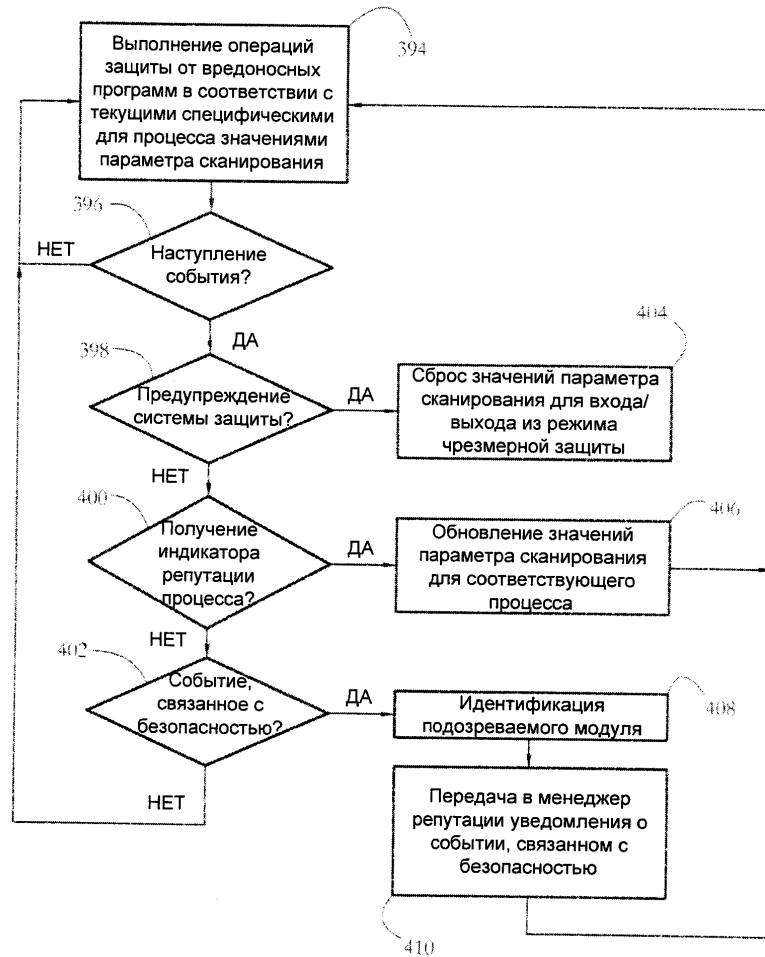
9/12



Фиг. 14



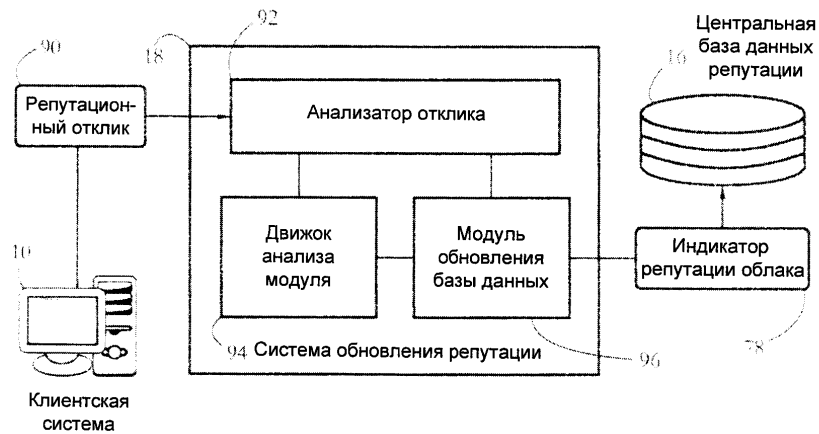
Фиг. 15



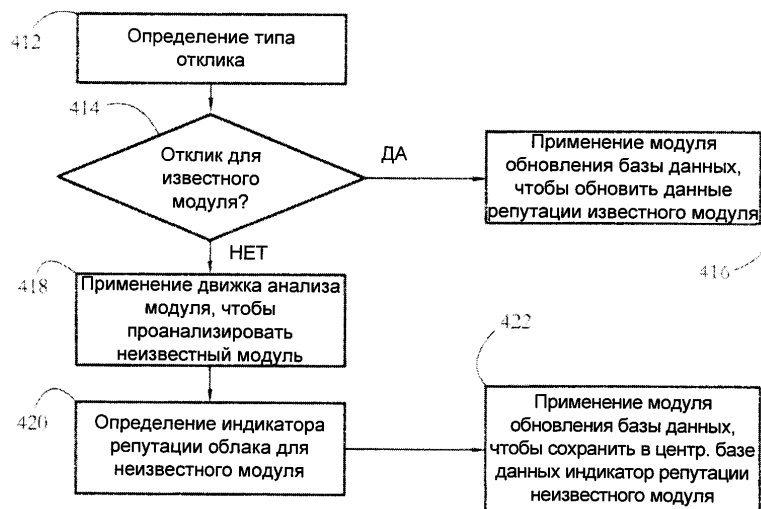
Фиг. 16

36586

12/12



Фиг. 17



Фиг. 18