РОССИЙСКАЯ ФЕДЕРАЦИЯ

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(19) **RU** (11) **2 645 265**(13) **C2**

(51) МПК
G06F 3/048 (2013.01)
G06F 21/00 (2013.01)

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК
G06F 3/048 (2017.05); G06F 21/554 (2017.05); G06F 21/566 (2017.05)

(21)(22) Заявка: **2013153762, 05.12.2013**

(24) Дата начала отсчета срока действия патента:
05.12.2013

Дата регистрации:
19.02.2018

Приоритет(ы):
(22) Дата подачи заявки: **05.12.2013**

(43) Дата публикации заявки: **10.06.2015** Бюл. № **16**

(45) Опубликовано: **19.02.2018** Бюл. № **5**

Адрес для переписки:
125212, Москва, Ленинградское ш., 39а, стр. 3, АО "Лаборатория Касперского", Управление по интеллектуальной собственности, Надежда Васильевна Кащенко

(72) Автор(ы):
Филатов Константин Михайлович (RU),
Яблоков Виктор Владимирович (RU)

(73) Патентообладатель(и):
Закрытое акционерное общество "Лаборатория Касперского" (RU)

(56) Список документов, цитированных в отчете о поиске: .

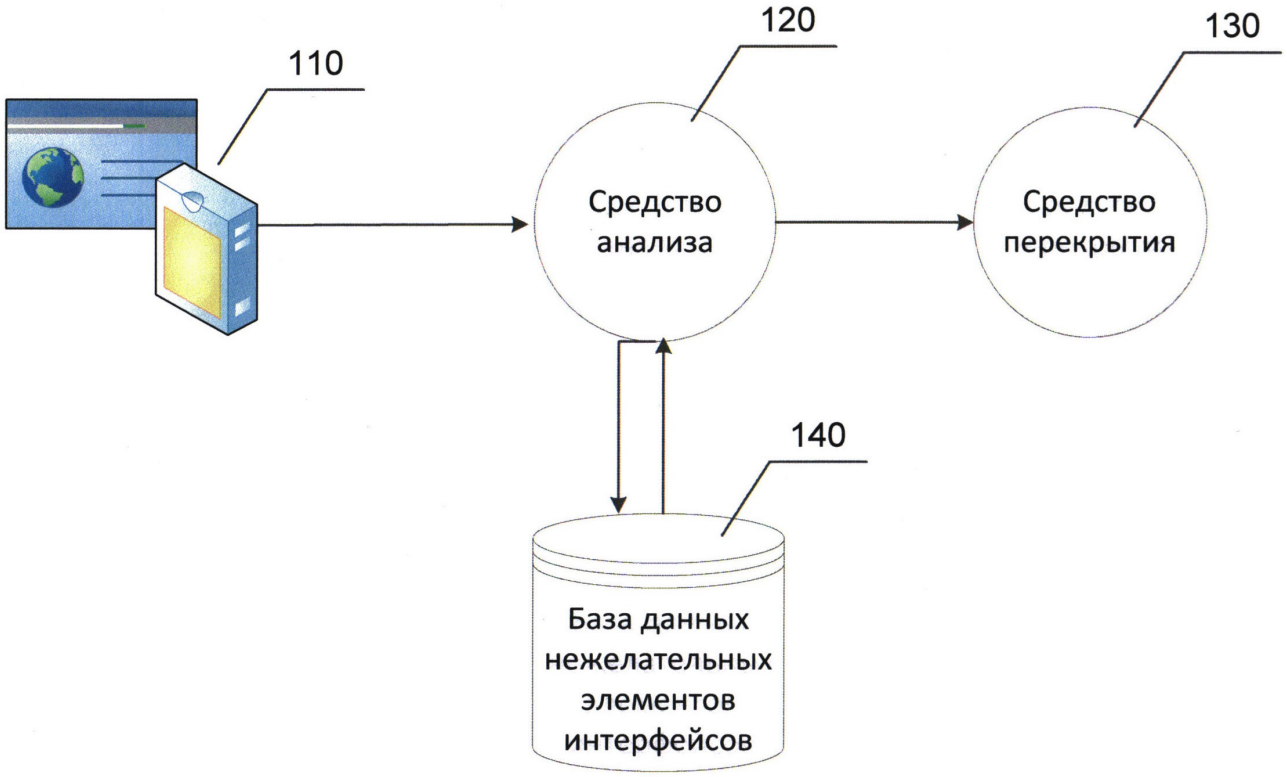
(54) Система и способ блокировки элементов интерфейса приложения

(57) Реферат:

Изобретение относится к системе и способу ограничения доступа к приложениям. Технический результат настоящего изобретения заключается в ограничении доступа к интерфейсу нежелательного приложения. Указанный технический результат достигается за счет блокировки нежелательного элемента интерфейса активного приложения путем его перекрытия. При перекрытии отрисовывают новый графический элемент поверх нежелательного элемента интерфейса активного приложения мобильного устройства. 2 н.п. ф-лы, 3 ил.

RU 2645265 C2

RU 2645265 C2



Фиг. 1

RU 2645265 C2

RU 2645265 C2

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 645 265**(13) **C2**

(51) Int. Cl.
G06F 3/048 (2013.01)
G06F 21/00 (2013.01)

(12) ABSTRACT OF INVENTION

(52) CPC
G06F 3/048 (2017.05); G06F 21/554 (2017.05); G06F 21/566 (2017.05)

(21)(22) Application: **2013153762, 05.12.2013**

(24) Effective date for property rights:
05.12.2013

Registration date:
19.02.2018

Priority:
(22) Date of filing: **05.12.2013**

(43) Application published: **10.06.2015** Bull. № **16**

(45) Date of publication: **19.02.2018** Bull. № **5**

Mail address:
125212, Moskva, Leningradskoe sh., 39a, str. 3, AO "Laboratoriya Kasperskogo", Upravlenie po intellektualnoj sobstvennosti, Nadezhda Vasilevna Kashchenko

(72) Inventor(s):
Filatov Konstantin Mikhajlovich (RU),
Yablokov Viktor Vladimirovich (RU)

(73) Proprietor(s):
Zakrytoe aktsionernoe obshchestvo "Laboratoriya Kasperskogo" (RU)

RU 2645265 C2

(54) SYSTEM AND METHOD OF BLOCKING ELEMENTS OF APPLICATION INTERFACE

(57) Abstract:

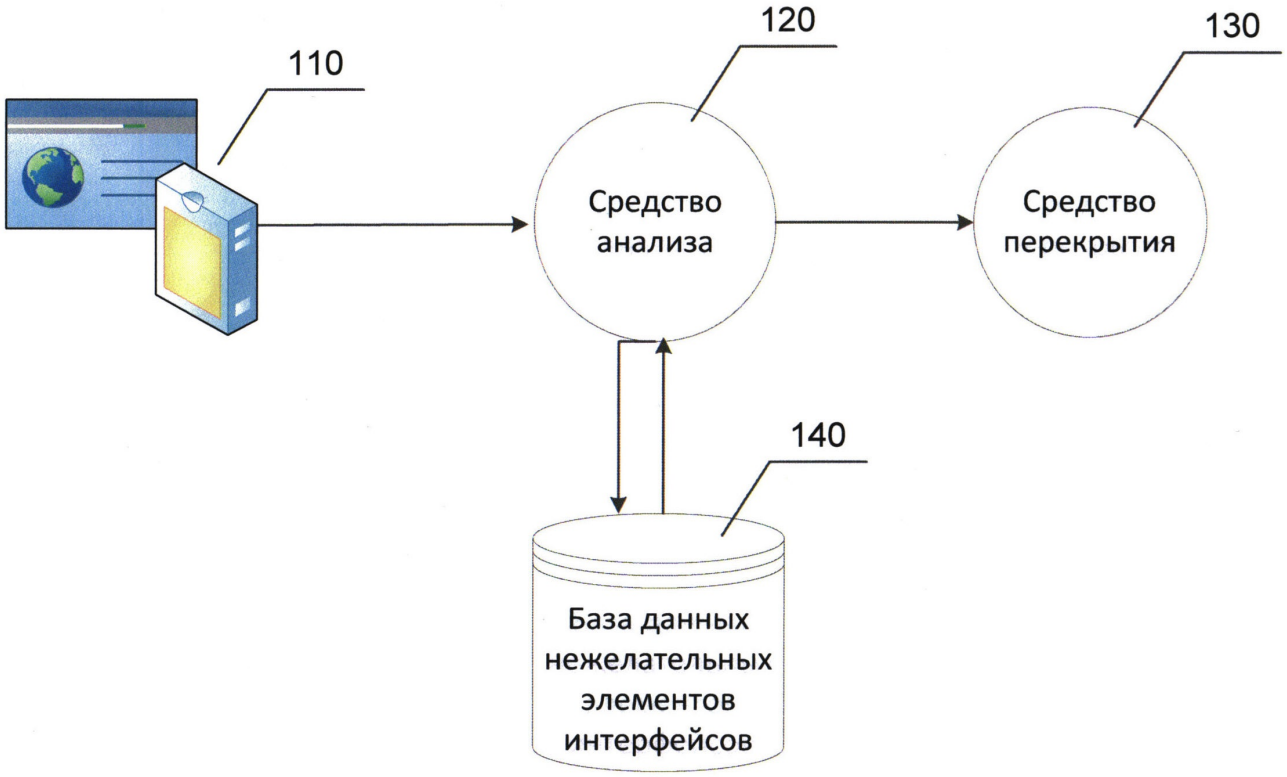
FIELD: information technology.

SUBSTANCE: method is implemented by blocking the unwanted element of the active application's interface by overlapping it. When overlapping, new graphic element is drawn on top of the unwanted element of the active application's interface of mobile device.

EFFECT: restricting access to the interface of unwanted application.

2 cl, 3 dwg

RU 2645265 C2



Фиг. 1

RU 2645265 C2

Область техники

Изобретение относится к системам и способам ограничения доступа к функциям приложений.

Уровень техники

В настоящее время количество программного обеспечения (ПО), с которым работает пользователь, постоянно возрастает. Современное ПО в большинстве случаев имеет интерфейс, который необходим для взаимодействия пользователя с ПО.

Интерфейс - упорядоченное отображение функции ПО, в котором представлены элементы управления: меню, командные строки, кнопки, значки, списки и т.д., исполненные в виде графических изображений. Пользователь имеет произвольный доступ (с помощью устройств ввода - клавиатуры, мыши, джойстика, сенсорного экрана и т.д.) ко всем видимым экранным объектам (элементам интерфейса) и осуществляет непосредственное манипулирование ими. Чаще всего элементы интерфейса реализованы на основе метафор и отображают их назначение и свойства, что облегчает понимание и освоение приложений неподготовленными пользователями.

Возникают ситуации, например, при использовании родительского контроля, когда полное ограничение доступа к приложению не оправдано. Известно, что в большинстве современных игровых приложений необходимо оплачивать внутри-игровые покупки (in app purchase). Может возникнуть ситуация, когда родители разрешают ребенку использовать игровое приложение, но не разрешают тратить реальные деньги на внутри-игровые покупки. В этом случае ограничение доступа ко всему приложению будет не оправдано.

В настоящее время существует ряд решений, предназначенных для анализа взаимодействия пользователя и ПО через интерфейс. В заявке WO 2012176365A1 описана процедура замены одного изображения экрана другим при помощи модуля построения интерфейса внутри одного приложения. Сборка интерфейса выполняется на основании набора атрибутов компонентов. В публикации US 20080148235 A1 отмечен алгоритм, по которому система анализа интерфейса дает оценку интерфейсов приложения и сравнивает их с условиями дизайна, утвержденными пользователем, с целью определить, правильно ли отображаются элементы интерфейса приложения.

Указанные решения осуществляют анализ взаимодействия пользователя с ПО через интерфейс, но не ограничивают доступ пользователя к интерфейсу ПО в случае необходимости.

Настоящее изобретение позволяет эффективно решить задачу ограничения взаимодействия пользователя и приложения через интерфейс.

Раскрытие изобретения

Изобретение относится к системам и способам ограничения доступа к функциям приложений. Технический результат настоящего изобретения заключается в ограничении доступа к интерфейсу нежелательного приложения. Указанный технический результат достигается за счет блокировки нежелательного элемента интерфейса активного приложения путем его перекрытия. При перекрытии отрисовывают новый графический элемент поверх нежелательного элемента интерфейса активного приложения мобильного устройства.

Система блокировки элементов интерфейса приложений, которая содержит: по крайней мере, одно активное приложение, которое имеет интерфейс; средство анализа, предназначенное для определения факта отображения, по крайней мере, одного элемента интерфейса активного приложения мобильного устройства, определения нежелательности отображенного элемента интерфейса активного приложения

мобильного устройства путем сравнения отображенного элемента активного приложения с известными нежелательными элементами интерфейсов приложений из базы данных нежелательных элементов интерфейсов, при обнаружении нежелательного элемента интерфейса приложения мобильного устройства передачи информации о нежелательном элементе интерфейса активного приложения мобильного устройства средству перекрытия; базу данных нежелательных элементов интерфейсов, предназначенная для хранения образцов и параметров известных нежелательных элементов интерфейсов приложений; средство перекрытия, предназначенное для блокировки нежелательного элемента интерфейса приложения мобильного устройства путем его перекрытия.

В частном случае реализации системы средство перекрытия при перекрытии отрисовывает новый графический элемент поверх нежелательного элемента интерфейса активного приложения.

В другом частном случае реализации системы средство перекрытия при перекрытии использует элемент интерфейса другого приложения мобильного устройства.

Еще в одном частном случае реализации системы элемент интерфейса другого приложения является элемент интерфейса антивирусной программы.

В частном случае реализации системы средство анализа определяет факт отображения, по крайней мере, одного элемента интерфейса приложения мобильного устройства в синхронном режиме.

В другом частном случае реализации системы средство анализа определяет факт отображения, по крайней мере, одного элемента интерфейса приложения мобильного устройства в асинхронном режиме.

Еще в одном частном случае реализации системы средство анализа предназначено для определения факта выполнения действий пользователей над элементами интерфейса приложения мобильного устройства.

В частном случае реализации системы база данных нежелательных элементов интерфейсов предназначена для хранения образцов и шаблонов известных нежелательных действий.

В другом частном случае реализации системы средство перекрытия временно осуществляет перекрытие отображенного нежелательного элемента интерфейса приложения мобильного устройства.

Способ блокировки элементов интерфейса приложений, в котором: при помощи средства анализа определяют факт отображения, по крайней мере, одного элемента интерфейса активного приложения мобильного устройства; при помощи средства определяют нежелательность отображенного элемента интерфейса активного приложения мобильного устройства путем сравнения отображенного элемента активного приложения мобильного устройства с известными нежелательными элементами интерфейсов приложений; при выявлении нежелательного элемента интерфейса при помощи средства перекрытия блокируют нежелательный элемент интерфейса активного приложения мобильного устройства путем его перекрытия.

В частном случае реализации способа при перекрытии отрисовывают новый графический элемент поверх нежелательного элемента интерфейса активного приложения мобильного устройства.

В другом частном случае реализации способа при перекрытии используют элемент интерфейса другого приложения мобильного устройства.

Еще в одном частном случае реализации способа элементом интерфейса другого приложения мобильного устройства является элемент интерфейса антивирусной

программы.

В частном случае реализации способа при помощи средства анализа определяют факт отображения, по крайней мере, одного элемента интерфейса приложения мобильного устройства в синхронном режиме.

В другом частном случае реализации способа при помощи средства анализа определяют факт отображения, по крайней мере, одного элемента интерфейса приложения мобильного устройства в асинхронном режиме.

Еще в одном частном случае реализации способа при помощи средства анализа определяют факт выполнения действий пользователей над элементами интерфейса приложения мобильного устройства.

В частном случае реализации способа хранят образцы и шаблоны известных нежелательных действий.

Еще в одном частном случае реализации способа при помощи средства перекрытия временно осуществляют перекрытие отображенного нежелательного элемента интерфейса приложения мобильного устройства.

Краткое описание чертежей

Дополнительные цели, признаки и преимущества настоящего изобретения будут очевидными из прочтения последующего описания осуществления изобретения со ссылкой на прилагаемые чертежи, на которых:

Фиг.1 отображает структурную схему системы блокировки элементов интерфейса.

Фиг.2 иллюстрирует алгоритм работы системы блокировки элементов интерфейса.

Фиг.3 представляет пример компьютерной системы общего назначения.

Хотя изобретение может иметь различные модификации и альтернативные формы, характерные признаки, показанные в качестве примера на чертежах, будут описаны подробно. Следует понимать, однако, что цель описания заключается не в ограничении изобретения конкретным его воплощением. Наоборот, целью описания является охват всех изменений, модификаций, входящих в рамки данного изобретения, как это определено в приложенной формуле.

Описание вариантов осуществления изобретения

Объекты и признаки настоящего изобретения, способы для достижения этих объектов и признаков станут очевидными посредством отсылки к примерным вариантам осуществления. Однако, настоящее изобретение не ограничивается примерными вариантами осуществления, раскрытыми ниже, оно может воплощаться в различных видах. Сущность, приведенная в описании, является ничем иным, как конкретными деталями, необходимыми для помощи специалисту в области техники в исчерпывающем понимании изобретения, и настоящее изобретение определяется в объеме приложенной формулы.

Современное ПО имеет интерфейс, который обеспечивает удобный доступ к основным функциям ПО. Активное приложение отображает свой интерфейс на передний план устройства вывода (монитор, экран мобильного телефона и т.д.) Под интерфейсом понимают совокупность элементов управления функциями ПО, например, графический пользовательский интерфейс (ГПИ) ПО. Операционная система (ОС) - разновидность ПО, которое также имеет свой интерфейс. Элемент интерфейса - составная часть интерфейса, с помощью которой осуществляется диалог пользователя и ПО, например: окно, кнопка, полоса прокрутки, флажок, ссылка, иконка и т.д. Элементы интерфейса являются частью конкретного ПО и могут быть уникальными (панели инструментов, управления, отдельные окна и т.д.)

Пользователь в ходе работы с приложениями взаимодействует с их интерфейсами.

Активное приложение отображает свой интерфейс 110 на переднем плане. Если содержимое интерфейса или выполнение действий над его элементами нежелательно, а заблокировать или остановить работу приложения или отображение его интерфейса крайне затруднительно (например, когда недостаточно привилегий для закрытия приложения), можно ограничить работу пользователя с элементами интерфейса путем перекрытия элементов интерфейса активного приложения, например, рисунком (графическим элементом), элементами интерфейса доверенного приложения. Рисунок может быть отображен с использованием графических функций, реализованных в конкретной ОС. Рисунок постоянно отображается поверх нежелательного элемента интерфейса приложения. Рисунком в зависимости от размера нежелательного элемента интерфейса приложения может быть, например, черный или белый квадрат:

```
case WM_PAINT:
{
HDC hDC=::GetDC(NULL)
::Rectangle(hDC, 500, 500, 600, 600);
::ReleaseDC(NULL, hDC);
}
break;
```

Доверенным приложением может быть, например, антивирусное приложение. Элементом интерфейса доверенного приложения, которым будет перекрыт нежелательный элемент интерфейса активного приложения, может быть окно предупреждения о невозможности продолжения дальнейшей работы с приложением.

Под нежелательными элементами интерфейса понимают элементы интерфейса ПО, которые необходимо перекрыть. Примером нежелательных элементов интерфейса могут быть элементы интерфейса приложения, которое относится к определенной категории. Приложением, которое относится к категории запрещенных, например, может быть любая обнаруженная вредоносная программа. Другим примером нежелательных элементов интерфейса могут быть элементы интерфейса приложения, которые позволяют получить доступ к конфиденциальной информации, к тематической информации, просмотр которой ограничен, например, по возрасту. Также примером элементов нежелательного интерфейса приложения могут быть элементы интерфейса приложения, которые позволяют выполнять платные функции. Элементы интерфейса в вышеупомянутых приложениях могут, например, позволять отсылать SMS либо входить в сеть «Интернет» и т.д., что может быть нежелательным, поскольку подобные услуги могут быть платными. Также нежелательность элементов интерфейса может быть определена на основании обратной связи или ретроспективного анализа. В этом случае элементы интерфейса, через которые было осуществлено определенное действие (отослано платное SMS, запрос по интернету, запрос на определение координат), впоследствии анализируют и принимают решение об их нежелательности.

Для решения вышеописанной проблемы используют систему блокировки элементов интерфейса. На Фиг.1 отображена структурная схема системы блокировки элементов интерфейса. Система блокировки элементов интерфейса состоит из средства анализа 120, базы данных нежелательных элементов интерфейсов 140, средства перекрытия 130.

Средство анализа 120 предназначено для определения факта отображения, по крайней мере, одного элемента интерфейса активного приложения 110 на устройстве вывода, определения нежелательности отображенного элемента интерфейса активного приложения 110 путем сравнения отображенного элемента активного приложения 110

с известными нежелательными элементами интерфейсов приложений из базы данных нежелательных элементов интерфейсов 140. Также средство анализа предназначено для передачи информации о нежелательном элементе интерфейса активного приложения средству перекрытия 130 при обнаружении нежелательного элемента интерфейса приложения.

Видимая активность приложения начинается с отображения пользователю элементов его интерфейса 110. Средство анализа 120 определяет факт отображения элемента интерфейса приложения 110. Определение факта отображения элемента интерфейса приложения 110 может быть выполнено как в синхронном, так и в асинхронном режиме. В синхронном режиме средство анализа 120 определяет факт отображения элементов интерфейса приложений непосредственно после их отображения. В асинхронном режиме средство анализа 120 определяет факты отображения элементов интерфейса приложений с задержкой. Определение факта отображения элемента интерфейса приложения может быть выполнено, например, на основе системного журнала событий или через перехват сообщений системы. Также можно осуществлять анализ активных окон, например: на платформе Symbian существует класс RWindowGroup, который содержит метод EnableFocusChangeEvents. Далее средство анализа 120 осуществляет определение нежелательности отображенного элемента интерфейса активного приложения 110 путем сравнения отображенного элемента активного приложения 110 с известными нежелательными элементами интерфейсов приложений из базы данных нежелательных элементов интерфейсов 140. При обнаружении нежелательного элемента интерфейса приложения средство анализа 120 передает информацию о нежелательном элементе интерфейса активного приложения средству перекрытия 130.

В одном из вариантов реализации база данных нежелательных элементов интерфейсов 140 предназначена для хранения образцов и параметров известных нежелательных элементов интерфейсов приложений. Средство анализа 120 в ходе анализа может выполнять сравнение образцов и параметров отображенного элемента интерфейса приложения и известных нежелательных элементов интерфейсов из базы данных нежелательных элементов интерфейсов 140. Например, основным параметром сравнения может быть идентификатор окна интерфейса, а элементом сравнения - диалоговое окно. Помимо этого нежелательность элементов интерфейса может выполняться путем контентного анализа содержимого элемента интерфейса. В ходе анализа осуществляют поиск нежелательных ссылок (вредоносных, недетского содержания), анализ рисунков путем распознавания образов (порнография, конфиденциальная информация)

В качестве базы данных нежелательных элементов интерфейсов 140 может использоваться различные виды баз данных, а именно: иерархические (IMS, TDMS, System 2000), сетевые (Cerebrum, Cronospro, DBVist), реляционные (DB2, Informix, Microsoft SQL Server), объектно-ориентированные (Jasmine, Versant, POET), объектно-реляционные (Oracle Database, PostgreSQL, FirstSQL/J, функциональные и т.д.

В другом варианте реализации средство анализа 120 может определять факт выполнения конкретных действий пользователей над элементами интерфейса приложения. В этом случае база данных нежелательных элементов интерфейсов 140 предназначена для хранения образцов и шаблонов известных нежелательных действий. В свою очередь средство анализа 120 определяет факт выполнения действия над элементом интерфейса, а затем в ходе анализа осуществляет сравнение выполненного действия пользователя над элементами интерфейса приложения с известными нежелательными действиями из базы данных элементов интерфейсов 140. При обнаружении нежелательного действия средство анализа 120 передает информацию о

нежелательном действии средству перекрытия 130.

Нежелательное действие - действие над элементом интерфейса приложения, которое не должно быть выполнено. Примером нежелательного действия может быть запуск приложения, которое относится к определенной категории. Под запуском можно рассматривать, например, двойной щелчок мыши по элементу интерфейса в виде иконки на рабочем столе ОС Microsoft Windows. Приложением, которое относится к категории запрещенных, например, может быть любая детектируемая вредоносная программа. Поэтому любой вариант запуска вредоносной программы можно считать нежелательным действием.

Нежелательность действия включает в себя, по крайней мере, три составляющих. Первой составляющей является действие, которое совершается. Некоторые действия, совершаемые пользователем при работе с элементами интерфейса, можно обобщить в группы. В основную группу входят действия пользователя, направленные на запуск приложения. Запуск может быть осуществлен несколькими вариантами в зависимости от операционной системы и приложения. Например, в интерфейсе ОС Windows запуск может быть осуществлен двойным щелчком по элементу интерфейса (иконка), одним щелчком по элементу интерфейса (ссылка), клавишей «Enter», сочетанием клавиш, щелчком левой кнопки мыши по элементу контекстного меню «открыть», вызываемого правой кнопкой мыши. В консоли, например, запуск может быть осуществлен набором пути к исполняемому файлу приложения и нажатием клавиши «Enter». В операционной системе Android запуск приложения осуществляется нажатием на сенсорном экране.

Второй составляющей нежелательности действия является элемент интерфейса, над которым совершают действие. Нежелательным может быть нажатие на определенную кнопку, например, на кнопку «отправить». Также объектом может быть строка контекстного меню, флажок т.д.

Третьей составляющей нежелательного действия является причина, по которой действие является нежелательным. Причина нежелательности действия может зависеть от репутации элемента интерфейса. Например, когда элементом интерфейса является ярлык приложения, которое находится в категории запрещенных. Таким образом, причина нежелательности может заключаться в принадлежности приложения к запрещенной категории. Корректно созданный ярлык в ОС Windows непосредственно связан с исполняемым файлом приложения. Следовательно, если приложение принадлежит к категории запрещенных, его ярлык не следует запускать. Поэтому нежелательным действием может быть двойной щелчок по ярлыку приложения, которое принадлежит к категории запрещенных. В другом случае объектом действия может быть кнопка, по нажатию которой, осуществляется переход или появляется окно, которое находится в категории запрещенных для просмотра. Примером этого может быть окно, в котором необходимо ввести номер карты и дополнительные данные, чтобы совершить оплату. Таким образом, нежелательным действием может быть нажатие на определенную кнопку.

Также причиной нежелательности может быть ценность информации, к которой после выполнения действия организуется доступ. В этом случае любые действия, которые приводят к открытию, например, текстовых файлов, могут быть нежелательными.

Еще одной причиной нежелательности могут быть нехарактерные для конкретного пользователя действия. Пользователь ранее определил порядок действий, который однозначно его идентифицирует. Например, в ОС Android пользователь может определить следующий порядок действий: разблокировка экрана, активация верхнего выпадающего меню, отключение вибрации телефона, включение вибрации телефона.

Любой пользователь, выполнив вышеупомянутый порядок действий, сможет использовать все возможности установленных приложений без ограничений. Если злоумышленник узнал пароль доступа к устройству, но не выполнил порядок действий, который однозначно идентифицирует пользователя, то интерфейсы приложений, которые предоставляют личные данные либо другие данные, доступ к которым нежелателен посторонним, будут перекрыты. Помимо перекрытия интерфейса, устройство, на котором не был выполнен вышеупомянутый порядок действий, может выслать оповещающее сообщение о том, что была осуществлена несанкционированная попытка доступа к личным данным.

Помимо отдельных действий пользователя, можно анализировать нежелательность совокупности действий над элементами интерфейса. Выявление всевозможных переходов по элементам интерфейса одного приложения можно описать при помощи сценария действий, выполняемых пользователем. Сценарий действий может состоять из нескольких переходов по элементам интерфейса одного приложения. Например, в случае, если окно интерфейса содержит кнопки перехода («Далее»). Использование сценариев действий может быть эффективным в тех случаях, когда нежелательные элементы интерфейса появляются только после выполнения нескольких переходов. Сценарии действий так же могут быть нежелательными в случае, если их выполнение приведет к отображению нежелательного интерфейса. В этом случае при выявлении достаточной степени сходства в текущих действиях пользователя над элементами интерфейса приложения с нежелательным сценарием действий следует также выполнить перекрытие элементов интерфейса.

Средство перекрытия 130 предназначено для блокировки нежелательного элемента интерфейса путем его перекрытия.

Блокировка нежелательного элемента интерфейса приложений осуществляется путем его перекрытия. Таким образом, пользователь не сможет получить доступ к нежелательному приложению и использовать его функции, поскольку не увидит никаких элементов интерфейса нежелательного приложения, а, следовательно, и взаимодействовать с ними для использования функций.

Если отображаемый элемент интерфейса активного приложения является нежелательным, то средство перекрытия 130 блокирует нежелательный элемент интерфейса путем его перекрытия. При перекрытии происходит отрисовка нового графического элемента поверх нежелательного элемента интерфейса активного приложения. Если отображаемый элемент интерфейса приложения не является нежелательным, то система переходит к анализу элементов интерфейса следующего приложения.

В случае если пользователь решит снова обратиться к приложению с нежелательным элементом интерфейса, например, развернуть приложение или повторно открыть его, нежелательный элемент интерфейса приложения снова будет перекрыт.

Перекрытие также может носить временный характер. Например, ограничение доступа к игровым приложениям может быть снято на период свободного времени ребенка, и возобновлено по его окончании.

Систему блокировки нежелательных элементов интерфейса эффективно использовать в мобильных ОС, где все приложения имеют одинаковый уровень привилегий, например, в ОС Android. В этом случае одно приложение не имеет возможности закрыть другое (ограничить выполнение). В тоже время реализация возможна в других операционных системах, например: Microsoft Windows, Symbian, Tizen. Например, в ОС Android работа системы блокировки нежелательных элементов интерфейса основана на поиске в

системном журнале событий, с помощью которых можно определить факт отображения элемента интерфейса приложения, и периодической проверки элементов интерфейса приложения, которое видит пользователь.

Поиск событий, с помощью которых можно определить факт отображения элемента интерфейса приложения, выполняется в системном журнале ОС Android. Журналы от всех приложений в ОС Android хранятся в едином хранилище - системном журнале Logcat. При запуске доверенного приложения выполняют подключение к системному журналу. В ходе работы пользователя журнал заполняется записями, которые подвергаются анализу. Особое внимание уделяют записям с подписью «activitymanager». ActivityManager - компонент ОС Android, который отвечает за создание и переключение элементов интерфейса приложений, а также за управление их жизненным циклом. Если запись с подписью «activitymanager» начинается со слов «Starting» или «Displayed», это означает, что пользователь видит элементы интерфейса приложения. Также в этой записи указывается название пакета приложения, которое было выведено на экран.

Периодическая проверка того, какое приложение сейчас отображено на экране:

final ActivityManager am=(ActivityManager)

mContext.getSystemService(Service.ACTIVITY_SERVICE);

final List<ActivityManager.RunningTaskInfo>tasks=am.getRunningTasks(1);

final ActivityManager.RunningTasklnfo task=tasks.get(0);

final String pkgName=task.topActivity.getPackageName();

task.topActivity - элемент интерфейса приложения, который отображается на экране в данный момент.

pkgName - имя пакета приложения, которое было выведено на экран.

Для определения нежелательных элементов интерфейса приложения можно использовать черные списки пакетов и элементов интерфейсов приложения. В этом случае происходит сравнение отображаемых элементов интерфейса активного приложения с элементами интерфейса приложений из черного списка. Если обнаружено совпадение, то соответствующий отображаемый элемент интерфейса активного приложения считается нежелательным. Фиг.2 иллюстрирует алгоритм работы системы блокировки элементов интерфейса. На этапе 210 активное приложение отображает, по крайней мере, один элемент интерфейса активного приложения 110. На этапе 211 средство анализа определяет факт отображения, по крайней мере, одного элемента интерфейса активного приложения 110. На этапе 212 средство анализа 120 определяет нежелательность отображенного элемента интерфейса активного приложения 110 путем сравнения отображенного элемента интерфейса активного приложения 110 с известными нежелательными элементами интерфейсов приложений из базы данных нежелательных элементов интерфейсов 140. В случае, если после определения элемент интерфейса активного приложения 110 не является нежелательным, на этапе 213 пользователь продолжает работу с приложением. На этапе 214 при обнаружении нежелательного элемента интерфейса активного приложения средство анализа 120 передает информацию о нежелательном элементе интерфейса активного приложения средству перекрытия 130. Средство перекрытия 130 блокирует нежелательный элемент интерфейса активного приложения путем его перекрытия.

Фиг.3 представляет пример компьютерной системы общего назначения, персональный компьютер или сервер 20, содержащий центральный процессор 21, системную память 22 и системную шину 23, которая содержит разные системные компоненты, в том числе память, связанную с центральным процессором 21. Системная шина 23 реализована, как любая известная из уровня техники шинная структура, содержащая в свою очередь

память шины или контроллер памяти шины, периферийную шину и локальную шину, которая способна взаимодействовать с любой другой шинной архитектурой. Системная память содержит постоянное запоминающее устройство (ПЗУ) 24, память с произвольным доступом (ОЗУ) 25. Основная система ввода/вывода (BIOS) 26, содержит основные процедуры, которые обеспечивают передачу информации между элементами персонального компьютера 20, например, в момент загрузки операционной системы с использованием ПЗУ 24.

Персональный компьютер 20 в свою очередь содержит жесткий диск 27 для чтения и записи данных, привод магнитных дисков 28 для чтения и записи на сменные магнитные диски 29 и оптический привод 30 для чтения и записи на сменные оптические диски 31, такие как CD-ROM, DVD-ROM и иные оптические носители информации. Жесткий диск 27, привод магнитных дисков 28, оптический привод 30 соединены с системной шиной 23 через интерфейс жесткого диска 32, интерфейс магнитных дисков 33 и интерфейс оптического привода 34 соответственно. Приводы и соответствующие компьютерные носители информации представляют собой энергонезависимые средства хранения компьютерных инструкций, структур данных, программных модулей и прочих данных персонального компьютера 20.

Настоящее описание раскрывает реализацию системы, которая использует жесткий диск 27, сменный магнитный диск 29 и сменный оптический диск 31, но следует понимать, что возможно применение иных типов компьютерных носителей информации 56, которые способны хранить данные в доступной для чтения компьютером форме (твердотельные накопители, флеш-карты памяти, цифровые диски, память с произвольным доступом (ОЗУ) и т.п.), которые подключены к системной шине 23 через контроллер 55.

Компьютер 20 имеет файловую систему 36, где хранится записанная операционная система 35, а также дополнительные программные приложения 37, другие программные модули 38 и данные программ 39. Пользователь имеет возможность вводить команды и информацию в персональный компьютер 20 посредством устройств ввода (клавиатуры 40, манипулятора «мышь» 42). Могут использоваться другие устройства ввода (не отображены): микрофон, джойстик, игровая консоль, сканнер и т.п. Подобные устройства ввода по своему обычаю подключают к компьютерной системе 20 через последовательный порт 46, который в свою очередь подсоединен к системной шине, но могут быть подключены иным способом, например, при помощи параллельного порта, игрового порта или универсальной последовательной шины (USB). Монитор 47 или иной тип устройства отображения также подсоединен к системной шине 23 через интерфейс, такой как видеоадаптер 48. В дополнение к монитору 47, персональный компьютер может быть оснащен другими периферийными устройствами вывода (не отображены), например, колонками, принтером и т.п.

Персональный компьютер 20 способен работать в сетевом окружении, при этом используется сетевое соединение с другим или несколькими удаленными компьютерами 49. Удаленный компьютер (или компьютеры) 49 являются такими же персональными компьютерами или серверами, которые имеют большинство или все упомянутые элементы, отмеченные ранее при описании существа персонального компьютера 20, представленного на Фиг.3. В вычислительной сети могут присутствовать также и другие устройства, например, маршрутизаторы, сетевые станции, пиринговые устройства или иные сетевые узлы.

Сетевые соединения могут образовывать локальную вычислительную сеть (LAN) 50 и глобальную вычислительную сеть (WAN).

Такие сети применяются в корпоративных компьютерных сетях, внутренних сетях компаний и, как правило, имеют доступ к сети Интернет. В LAN- или WAN-сетях персональный компьютер 20 подключен к локальной сети 50 через сетевой адаптер или сетевой интерфейс 51. При использовании сетей персональный компьютер 20 может использовать модем 54 или иные средства обеспечения связи с глобальной вычислительной сетью, такой как Интернет. Модем 54, который является внутренним или внешним устройством, подключен к системной шине 23 посредством последовательного порта 46. Следует уточнить, что сетевые соединения являются лишь примерными и не обязаны отображать точную конфигурацию сети, т.е. в действительности существуют иные способы установления соединения техническими средствами связи одного компьютера с другим.

(57) Формула изобретения

1. Система перекрытия элемента интерфейса активного приложения, которая содержит:

а) по крайней мере одно активное приложение, которое имеет интерфейс;

б) средство анализа, предназначенное для:

• определения факта отображения по крайней мере одного элемента интерфейса активного приложения на основе журнала событий и активных окон;

• определения нежелательного отображенного элемента интерфейса активного приложения путем анализа по крайней мере одного из нижеследующего:

- параметры элемента интерфейса активного приложения,

- содержимое элемента интерфейса активного приложения,

- действия пользователя над отображенным элементом интерфейса активного приложения,

- категория активного приложения, факт отображения интерфейса которого был определен,

- ценность информации, к которой при помощи элемента интерфейса активного приложения получают доступ;

• при определении нежелательного отображенного элемента интерфейса активного приложения передачи информации о нежелательном отображенном элементе активного приложения интерфейса средству перекрытия;

в) базу данных нежелательных элементов интерфейсов, предназначенную для хранения образцов и параметров известных нежелательных элементов интерфейсов приложений, образцов известных нежелательных действий пользователя над элементами интерфейса приложений;

г) средство перекрытия, предназначенное для перекрытия нежелательного отображенного элемента интерфейса активного приложения элементом интерфейса антивирусной программы.

2. Способ перекрытия элемента интерфейса активного приложения, в котором:

а) при помощи средства анализа определяют факт отображения по крайней мере одного элемента интерфейса активного приложения на основе журнала событий и активных окон;

б) при помощи средства анализа определяют нежелательный отображенный элемент интерфейса активного приложения путем анализа по крайней мере одного из нижеследующего:

• параметры элемента интерфейса активного приложения,

• содержимое элемента интерфейса активного приложения,

• действия пользователя над отображенным элементом интерфейса активного приложения,

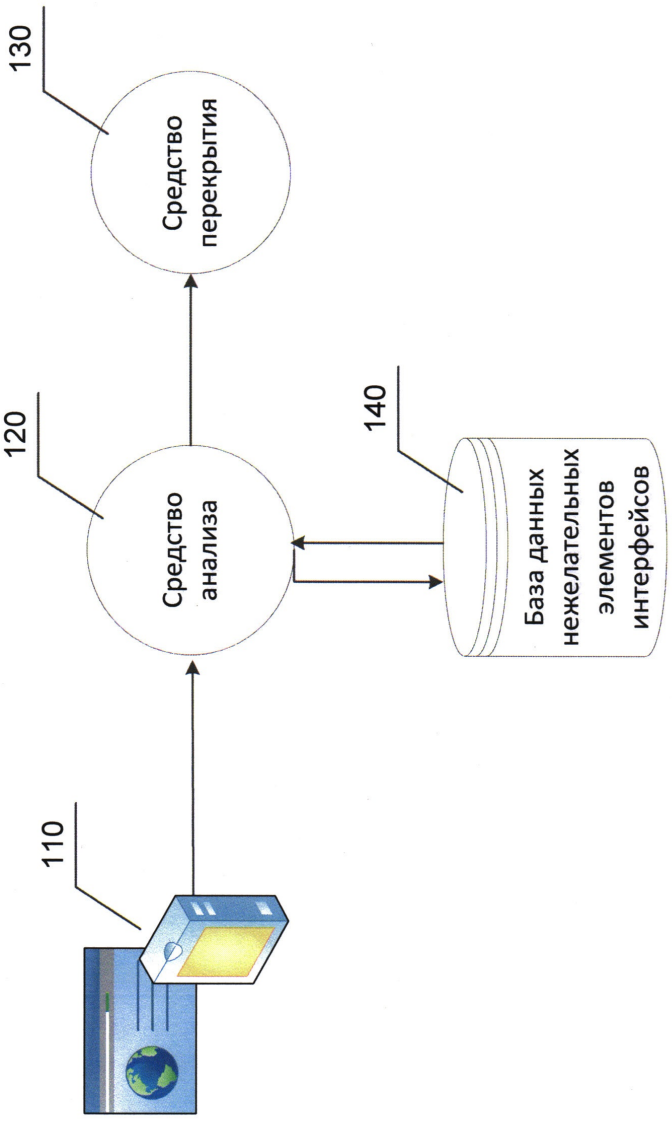
• категория активного приложения, факт отображения интерфейса которого был определен,

• ценность информации, к которой при помощи элемента интерфейса активного приложения получают доступ;

в) при помощи базы данных нежелательных элементов интерфейсов хранят образцы и параметры известных нежелательных элементов интерфейсов приложений, образцы известных нежелательных действий пользователя над элементами интерфейса приложений;

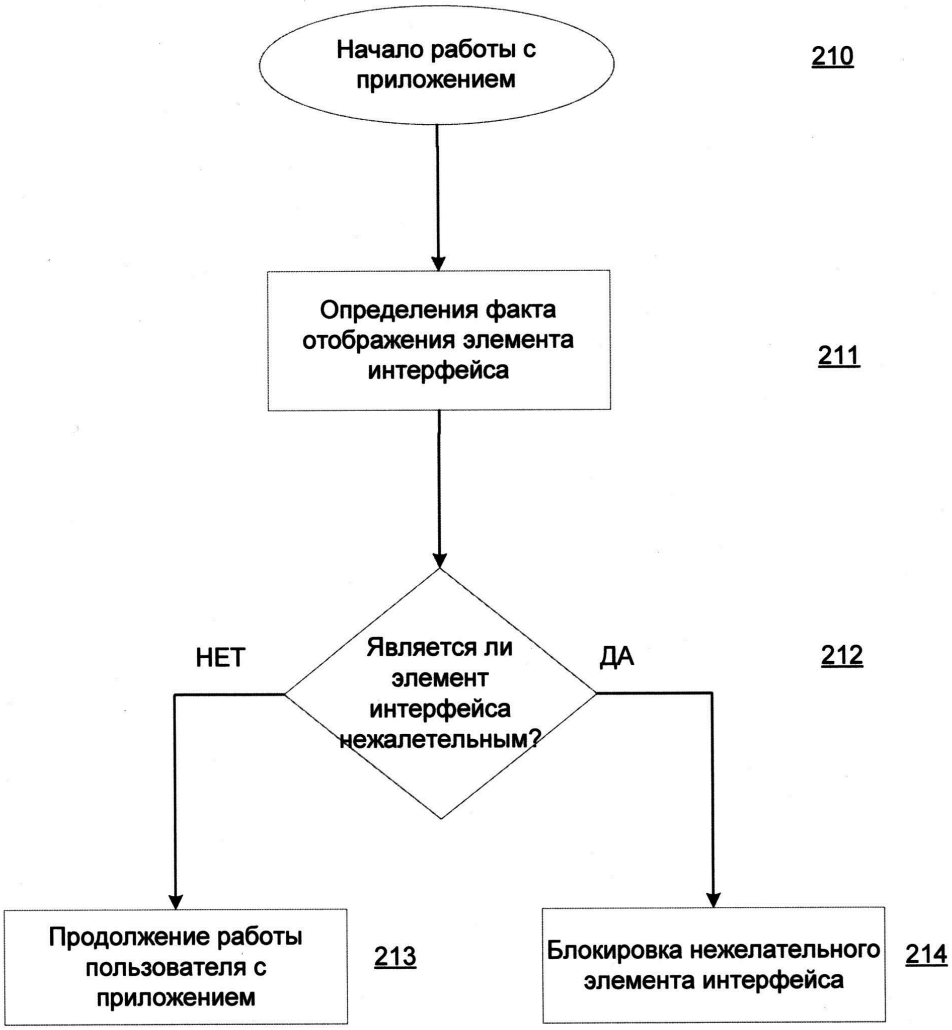
г) при помощи средства перекрытия перекрывают нежелательный отображенный элемент интерфейса приложения элементом интерфейса антивирусной программы.

1


130
Средство перекрытия
120
Средство анализа
140
База данных нежелательных элементов интерфейсов
110

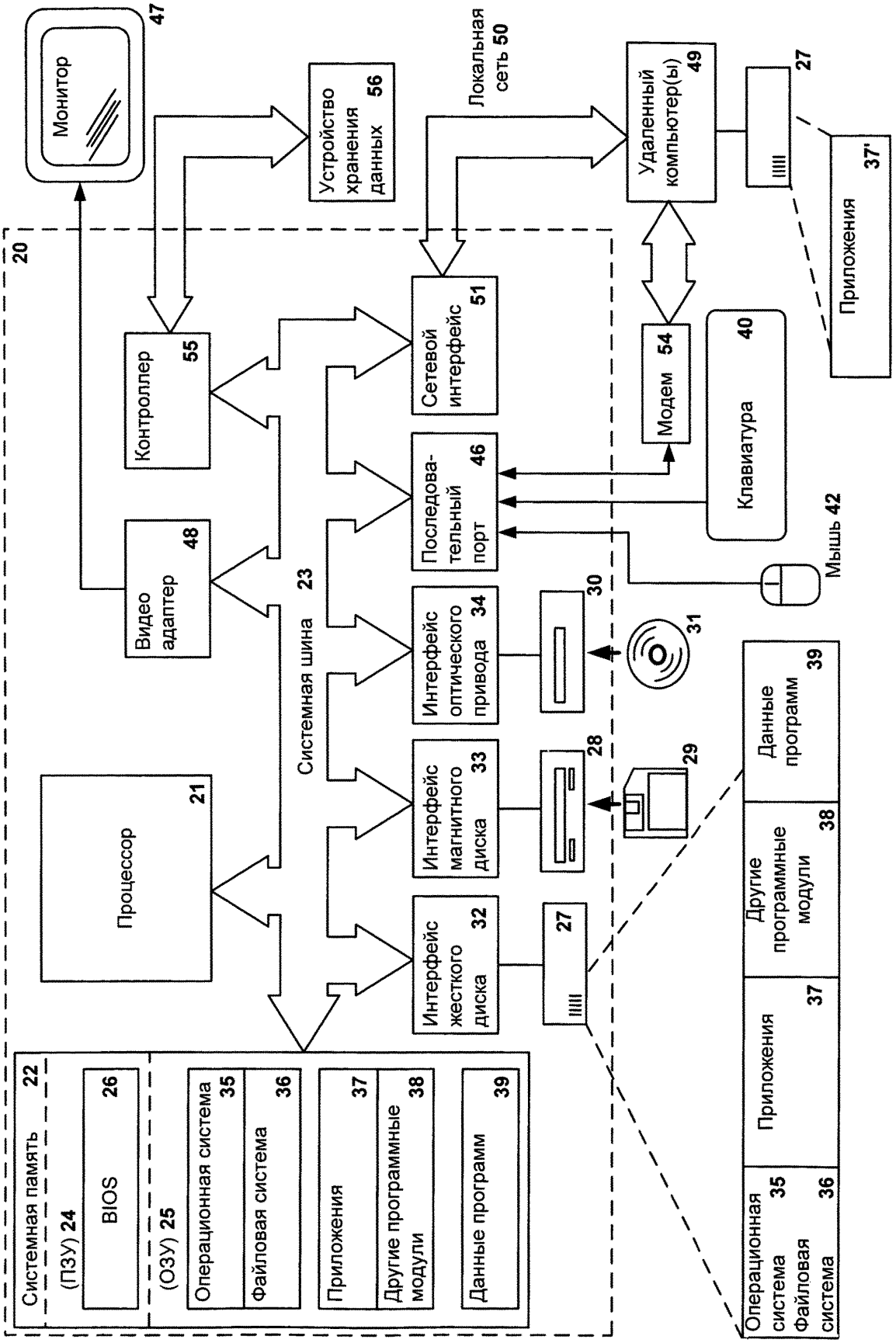
Фиг. 1

2


Начало работы с приложением
210
Определения факта отображения элемента интерфейса
211
Является ли элемент интерфейса нежалетельным?
212
НЕТ
ДА
Продолжение работы пользователя с приложением
213
Блокировка нежелательного элемента интерфейса
214

Фиг. 2

3


Монитор 47
Устройство хранения данных 56
Локальная сеть 50
Удаленный компьютер(ы) 49
27
Приложения 37'
20
Контроллер 55
Сетевой интерфейс 51
Модем 54
Клавиатура 40
Видео адаптер 48
Последовательный порт 46
Мышь 42
Системная шина 23
Интерфейс оптического привода 34
30
31
Процессор 21
Интерфейс магнитного диска 33
28
29
Данные программ 39
Другие программные модули 38
Интерфейс жесткого диска 32
27
Приложения 37
Операционная система 35
Файловая система 36
Системная память 22
(ПЗУ) 24
BIOS 26
(ОЗУ) 25
Операционная система 35
Файловая система 36
Приложения 37
Другие программные модули 38
Данные программ 39

Фиг. 3