РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 700 185**(13) **C1**

(51) МПК
G06F 21/55 (2013.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК
G06F 21/552 (2019.05)

(21)(22) Заявка: **2018127626, 27.07.2018**

(24) Дата начала отсчета срока действия патента:
27.07.2018

Дата регистрации:
13.09.2019

Приоритет(ы):
(22) Дата подачи заявки: **27.07.2018**

(45) Опубликовано: **13.09.2019** Бюл. № **26**

Адрес для переписки:
127287, Москва, Старый Петровско-Разумовский пр-д, 1/23, стр. 1, Закрытое акционерное общество "Перспективный мониторинг"

(72) Автор(ы):
Андрюхин Евгений Владимирович (RU)

(73) Патентообладатель(и):
Закрытое акционерное общество "Перспективный мониторинг" (RU)

(56) Список документов, цитированных в отчете о поиске: **RU 2624554 C1, 04.07.2017. RU 98613 U1, 20.10.2010. RU 98613 U1, 20.10.2010. RU 2454705 C1, 27.06.2012. US 2007/0300061 A1, 27.12.2007. US 7845009 B2, 30.11.2010. WO 2010/044616 A2, 22.04.2010.**

(54) Способ обнаружения скрытого программного обеспечения в вычислительной системе, работающей под управлением POSIX-совместимой операционной системы

(57) Реферат:

Изобретение относится к области обнаружения скрытого программного обеспечения в вычислительных системах, работающих под управлением POSIX-совместимых операционных систем. Техническим результатом является повышение защищенности вычислительной системы. В способе сравнивают значения статусов name, gid, полученные из 1-го программного средства, с соответствующими значениями name, gid, полученными из 4-го программного средства; сравнивают значения статусов name и uid, полученные из 2-го программного средства, с соответствующими значениями name и uid, полученными из 1-го программного средства; сравнивают значения статусов name и pid, полученные из 2-го программного средства, с соответствующими значениями name и pid, полученными из 3-го программного средства; сравнивают значения статусов pid и uid, полученные из 3-го программного средства, с соответствующими значениями pid и uid, полученными из 2-го программного средства; сравнивают значения статусов name, pid, ppid, state, полученные из 3-го программного средства, с соответствующими значениями name, pid, ppid, state, полученными из 4-го программного средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым; предоставляют отчет о наличии скрытых приложений и процессов.

RU 2700185 C1

RU 2700185 C1

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 700 185**(13) **C1**

(51) Int. Cl.
G06F 21/55 (2013.01)

(12) ABSTRACT OF INVENTION

(52) CPC
G06F 21/552 (2019.05)

(21)(22) Application: **2018127626, 27.07.2018**

(24) Effective date for property rights:
27.07.2018

Registration date:
13.09.2019

Priority:
(22) Date of filing: **27.07.2018**

(45) Date of publication: **13.09.2019** Bull. № **26**

Mail address:
127287, Moskva, Staryj Petrovsko-Razumovskij pr-d, 1/23, str. 1, Zakrytoe aktsionernoe obshchestvo "Perspektivnyj monitoring"

(72) Inventor(s):
Andryukhin Evgenij Vladimirovich (RU)

(73) Proprietor(s):
Zakrytoe aktsionernoe obshchestvo "Perspektivnyj monitoring" (RU)

(54) METHOD FOR DETECTING HIDDEN SOFTWARE IN A COMPUTING SYSTEM OPERATING UNDER A POSIX-COMPATIBLE OPERATING SYSTEM

(57) Abstract:

FIELD: computer equipment.

SUBSTANCE: invention relates to detection of hidden software in computer systems operating under POSIX-compatible operating systems. Method comprises comparing values of statuses name, gid, obtained from 1st software, with corresponding values of name, gid, obtained from 4-th software; comparing the values of statuses name and uid, obtained from 2nd software tool, with the corresponding values of name and uid, obtained from 1st software; comparing the values of the name and pid status values obtained from 2nd software tool with the corresponding values of name and pid, obtained from 3rd software; comparing the pid and uid status values obtained from 3rd software tool with the corresponding pid and uid values obtained from 2nd software; comparing the values of statuses name, pid, ppid, state, obtained from 3rd software tool, with corresponding values of name, pid, ppid, state, obtained from 4th software; if at least one of values of like states does not exist or does not match – application is considered to be hidden; report on availability of hidden applications and processes.

EFFECT: high security of the computer system.

1 cl

RU 2700185 C1

RU 2700185 C1

Область техники, к которой относится изобретение

Предлагаемое изобретение относится к вычислительной технике и, в частности, к способам обнаружения скрытого программного обеспечения в вычислительных системах, работающих под управлением POSIX-совместимых операционных систем, например, Solaris, Android и др.

Уровень техники

В современных вычислительных системах, работающих под управлением POSIX-совместимых операционных систем (ОС), возможно появление программ, предназначенных для сокрытия в системе определенных объектов либо активностей (руткитов). Чаще всего такая активность связана с работой вредоносного программного обеспечения (ПО). Сокрытию, как правило, подвергаются ключи реестра (например, отвечающие за автозапуск объектов), файлы, процессы в памяти зараженного компьютера, сетевая активность.

Существуют решения, направленные на обнаружение процессов или активностей в вычислительной системе и способные выявлять скрытые процессы, но не уведомлять об этом пользователя.

Так, например, в составе ОС Android имеются штатные программные средства Activity Manager, Usage Stats Manager, Package Manager, Procfs [Google Inc. Activity Manager. [Электронный ресурс] - Режим доступа к ресурсу: https://developer.android.com/reference/android/app/ActivityManager; Google Inc. Package Manager. [Электронный ресурс] - Режим доступа к ресурсу: https://developer.android.com/reference/android/content/pm/PackageManager; Google Inc. Usage Stats Manager. [Электронный ресурс] - Режим доступа к ресурсу: https://developer.android.com/reference/android/app/usage/UsageStatsManager; Procfs (Proc File System) [Электронный ресурс] - Режим доступа к ресурсу: https://ru.bmstu.wiki/Procfs_(Proc_File_System)].

Activity Manager является системным сервисом, обрабатывающим все активности, запущенные в системе. Простые приложения состоят из одной активности. Более сложные приложения могут иметь несколько окон, т.е. они состоят из нескольких активностей, которыми необходимо управлять и которые могут взаимодействовать между собой. Работает Activity Manager следующим образом: после запроса пользователя на запуск приложения, например, нажатием на соответствующую иконку в меню, срабатывает onClick() событие, вызывающее метод startActivity() у объекта Activity Manager. Поскольку приложение не может напрямую вызвать обработчик, оно формирует и отправляет запрос на Binder, который, в свою очередь, перенаправляет запрос на Service Manager и отправляет обработчик приложению [EastBanc Technologies. Об открытости данных в Android-приложениях. [Электронный ресурс] - Режим доступа к ресурсу: https://habr.com/company/eastbanctech/blog/212321/].

Таким образом, использование Activity Manager дает возможность контролировать список запущенных приложений, их имена и идентификаторы процессов.

Недостатком Activity Manager является возможность использования фоновых процессов, не имеющих активностей, что позволяет обойти контроль.

Usage Stats Manager является системным сервисом, предоставляющим доступ к статистике использования устройства, в том числе и активностями, функционирующими в системе. Работает Usage Stats Manager следующим образом: в момент времени, указанный системой, либо по запросу пользователя, имеющего политику безопасности «android.permission.PACKAGE_USAGE_STATS», в случайный момент времени, Usage Stats Manager осуществляет обращение к хранилищу внутренних событий ОС и статистики. В такие события входят запросы всех системных служб, изменения состояний

всех системных служб, а также выполненные действия.

Таким образом, использование Usage Stats Manager дает возможность контролировать список запущенных приложений, их состояния, имена, идентификаторы процессов и групп [Lorenzo Quiroli. Show app usage with UsageStatsManager. [Электронный ресурс] - Режим доступа к ресурсу: https://medium.com/@quiro91/show-app-usage-with-usagestatsmanager-d47294537dab].

Недостатком Usage Stats Manager является возможность модификации или повреждения внутреннего хранилища, что позволяет обойти контроль.

Package Manager работает следующим образом: информация об установленном приложении записывается в XML файл [Ketan Parmar. In Depth: Android Pack-age Manager and Package Installer. [Электронный ресурс] - Режим доступа к ресурсу: https://dzone.com/articles/depth-android-package-manager], затем Package Manager контролирует этот файл, добавляя и удаляя строки по мере установки и удаления пакетов. Получение списка процессов происходит путем чтения файла. Таким образом, использование Package Manager дает возможность контролировать список установленных приложений, а также просмотреть имя приложения, состояние и набор предоставленных приложению разрешений.

Недостатком Package Manager является возможность модификации или повреждения XML файла, что позволяет обойти контроль.

Программа Procfs позволяет получить доступ к информации о системных процессах, обрабатываемых в ядре ОС. При запуске Procfs создает двухуровневое представление пространств процессов. На верхнем уровне процессы представляют собой директории, именованные в соответствии с их идентификатором - pid. Также на верхнем уровне располагается ссылка на директорию, соответствующую процессу, выполняющему запрос; ссылка может иметь различное имя в различных ОС [Ross Anderson. Yet another Android side channel: input stealing for fun and profit. [Электронный ресурс] - Режим доступа к ресурсу: https://www.lightbluetouchpaper.org/2016/07/29/yet-another-android-side-channel/].

Недостатком Procfs является возможность модификации прошивки с целью внесения приложения в список системных, что позволяет обеспечить сокрытие приложения. Также современные версии ОС могут ограничивать доступ к Procfs со стороны процессов, запущенных с правами пользователя.

Известен также способ обнаружения скрытого программного обеспечения в вычислительной системе, работающей под управление POSIX-совместимой операционной системы [патент РФ №2624554, приоритет от 17.05.2016 г.], причем one-рационная система, помимо ядра, включает следующие программные средства:

- 1-е средство, выполненное с возможностью определять количество инсталлированных приложений в вычислительной системе,
- 2-е средство, выполненное с возможностью определять количество запущенных процессов в вычислительной системе,
- 3-е средство, выполненное с возможностью определять для процессов статусы pid, name, uid, groups, state,
- 4-е средство, выполненное с возможностью сравнивать результаты работы 1-го, 2-го и 3-го средств;

способ, заключающийся в том, что

- получают с помощью 1-го средства количество инсталлированных приложений в вычислительной системе;
- получают с помощью 2-го средства количество запущенных процессов в

вычислительной системе;

- получают с помощью 3-го средства значения статусов pid, name, uid, groups, state каждого процесса;
- выполняют с помощью 4-го средства для каждого процесса, список которых получен с помощью 3-го средства, следующие действия:
  - сравнивают значение статуса groups, полученное из 3-го средства, с нулем; если значение статуса groups равно нулю - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;
  - сравнивают значения статусов uid и name, полученные из 3-го средства, с соответствующими значениями uid и name, полученными из 1-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;
  - сравнивают значения статусов pid, name и state, полученные из 3-го средства, с соответствующими значениями pid, name и state, полученными из 2-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - процесс считается скрытым и сведения о процессе заносятся в отчет о наличии скрытых приложений и процессов;
- предоставляют отчет о наличии скрытых приложений и процессов;
- удаляют из вычислительной системы выявленные скрытые приложения.

Способ выбран в качестве прототипа.

В известном способе не учитываются значения статусов ppid, что является недостатком, который не позволяет обнаружить класс скрытого ПО, использующего механизм фреймирования [Android.Xiny.60 - Служба вирусного мониторинга Dr. Web. [Электронный ресурс] - Режим доступа к ресурсу: https://vms.drweb.ru/virus/?i=8624213&lng=ru]. Фреймирование позволяет скрыть процесс через интерфейс приложения, при этом во время запуска легитимных программ в основные процессы внедряется вредоносный модуль, который выполняет вредоносные функции от имени легитимного приложения.

Однако, значения статусов ppid и gid вредоносного модуля будут отличаться от значений легитимного приложения вследствие разности способов получения статусов процессов, но с использованием известного способа обнаружить это невозможно.

Это снижает защищенность вычислительной системы.

Раскрытие изобретения

Техническим результатом является повышение защищенности вычислительной системы.

Для этого предлагается способ обнаружения скрытого программного обеспечения в вычислительной системе, работающей под управлением POSIX-совместимой операционной системы, причем операционная система, помимо ядра, включает следующие программные средства:

- 1-е средство, выполненное с возможностью определять количество приложений, способных к запуску посредством операционной системы в вычислительной системе, их статусы name, gid, uid, а также индексировать сопровождающие их сервисы, службы и метаданные;
- 2-е средство, выполненное с возможностью определять количество запущенных процессов в вычислительной системе, а также их статусы name, pid, uid;
- 3-е средство, выполненное с возможностью определять для запущенных процессов статусы pid, ppid, name, uid, state;

● 4-е средство, выполненное с возможностью определять для запущенных процессов статусы gid, name, pid, ppid, state;

● 5-е средство, выполненное с возможностью сравнивать результаты работы 1-го, 2-го, 3-го и 4-го средств;

способ, заключающийся в том, что

● получают с помощью 1-го средства значения статусов name, gid, uid каждого приложения;

● получают с помощью 2-го средства значения статусов name, pid, uid каждого процесса;

● получают с помощью 3-го средства значения статусов pid, ppid, name, uid, state каждого процесса;

● получают с помощью 4-го средства значения статусов gid, name, pid, ppid, state каждого процесса;

● выполняют с помощью 5-го средства для каждого приложения, список которых получен с помощью 1-го средства, следующие действия:

○ сравнивают значения статусов name и gid, полученные из 1-го средства, с соответствующими значениями name и gid, полученными из 4-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

○ сравнивают значения статусов name и uid, полученные из 2-го средства, с соответствующими значениями name и uid, полученными из 1-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

○ сравнивают значения статусов name и pid, полученные из 2-го средства, с соответствующими значениями name и pid, полученными из 3-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

○ сравнивают значения статусов pid и uid, полученные из 3-го средства, с соответствующими значениями pid и uid, полученными из 2-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

○ сравнивают значения статусов name, pid, ppid, state, полученные из 3-го средства, с соответствующими значениями name, pid, ppid, state, полученными из 4-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

● предоставляют отчет о наличии скрытых приложений и процессов;

● при необходимости, удаляют из вычислительной системы выявленные скрытые приложения.

Для использования предложенного способа необходимо обеспечить в составе вычислительной системы следующие программные средства:

● 1-е средство, выполненное с возможностью определять количество приложений, способных к запуску посредством операционной системы в вычислительной системе, их статусы name, gid, uid, а также индексировать сопровождающие их сервисы, службы

и метаданные;

- 2-е средство, выполненное с возможностью определять количество запущенных процессов в вычислительной системе, а также их статусы name, pid, uid;
- 3-е средство, выполненное с возможностью определять для запущенных процессов статусы pid, ppid, name, uid, state;
- 4-е средство, выполненное с возможностью определять для запущенных процессов статусы gid, name, pid, ppid, state;
- 5-е средство, выполненное с возможностью сравнивать результаты работы 1-го, 2-го, 3-го и 4-го средств.

1-е средство отслеживает приложения, способные к запуску посредством операционной системы в вычислительной системе. Опытным путем установлено, что если приложение было установлено неофициальным путем (в обход официальных центров дистрибуции), то идентификаторы статусов, будут обрабатываться системой некорректно, что может быть проверено путем сравнения с помощью 5-го средства, группы статусов name-gid, полученной с помощью 1-го средства, с группой статусов name-gid, полученной с помощью 4-го средства.

В случае, если значения статусов name-gid не существуют или не совпали, то приложение считается скрытым, а факт сокрытия заносится в отчет.

2-е средство отслеживает запущенные активности.

Данные, полученные от 2-го средства, позволяют с помощью 5-го средства провести две проверки:

- во-первых, активность должна быть запущена приложением, следовательно, содержать запись о приложении в 1-ом средстве, что может быть проверено путем сравнения пары name-uid, полученной с помощью 2-го средства, с парой name-uid, полученной с помощью 1-го средства;
- во-вторых, информация о запущенной активности должна вноситься в хранилище внутренних событий ОС, что может быть проверено путем сравнения пары name-pid, полученной с помощью механизма 2-го средства, с парой name-pid, полученной с помощью механизма 3-го средства;

В случае если хотя бы одна из проверок неуспешна (т.е. соответствующие поля не совпали или хотя бы одно из них не определено), то приложение считается скрытым, а факт сокрытия заносится в отчет.

3-е средство получает данные из хранилища внутренних событий ОС.

Данные, полученные от 3-го средства, позволяют с помощью 5-го средства провести две проверки:

- во-первых, активность, содержащаяся в архиве, должна иметь запись об активности во 2-ом средстве, что может быть проверено путем сравнения статуса state, полученного с помощью 3-го средства, со значением «R» («Running»), а также пары статусов name-pid, полученной с помощью 3-го средства, с парой name-pid, полученной с помощью 2-го средства;
- во-вторых, запущенное приложение должно иметь активность, следовательно, имеет pid и состояние «Running», что может быть проверено путем сравнения группы name-pid-ppid-state, полученной с помощью 3-го средства с группой name-pid-ppid-state, полученной с помощью 4-го средства.

4-е средство получает данные о системных процессах из ядра операционной системы.

Данные, полученные от 4-го средства, позволяют с помощью 5-го средства провести две проверки:

- во-первых, необходимо определить приложения, установленные неофициальным

путем, что возможно проверить путем сравнения группы статусов name-gid, полученной с помощью 4-го средства, с группой статусов name-gid, полученной с помощью 1-го средства;

- во-вторых, приложение, получившее квант времени для выполнения кода на процессоре, должно быть отмечено как запущенное, что может быть проверено путем сравнения статуса state, полученного с помощью 4-го средства, со значением «R» («Running»), а также группы name-pid-ppid-state, полученной с помощью 4-го средства с группой name-pid-ppid-state, полученной с помощью 3-го средства. В случае если хотя бы одна из проверок неуспешна (т.е. соответствующие поля не совпали или хотя бы одно из них не определено), то приложение считается скрытым, а факт сокрытия заносится в отчет.

Таким образом, предложенный способ позволяет выявить скрытые приложения, которые известными способами не могут быть выявлены, и, при необходимости, предпринять меры по удалению (нейтрализации) этих приложений, что повышает защищенность вычислительной системы.

Осуществление изобретения

Реализация предложенного способа может быть осуществлена в вычислительной системе, работающих под управлением любой POSIX-совместимой ОС, например, Solaris, Android и др.

Рассмотрим осуществление способа на примере ОС Android 6.0.

В качестве 1-го средства, способного определять количество приложений, способных к запуску посредством операционной системы в вычислительной системе, может быть использована штатная программа Package Manager.

В качестве 2-го средства, способного определять количество запущенных процессов в вычислительной системе, а также их статусы name, pid, uid может быть использована штатная программа Activity Manager.

В качестве 3-го средства, способного определять для процессов статусы pid, ppid, gid, может быть использована штатная программа Usage Stats Manager.

В качестве 4-го средства, способного определять для процессов статусы gid, name, pid, ppid, state, uid, может быть использована штатная программа Procfs.

Для подготовки к использованию предлагаемого способа необходимо сформировать 5-е средство, позволяющее сравнивать результаты работы 1-го, 2-го, 3-го и 4-го средств.

Это средство представляет собой программу, которую, зная ее назначение и выполняемые функции, может сформировать специалист в области программирования (программист). Подготовленное 5-е средство после сформирования устанавливается (инсталлируется) в вычислительную систему.

Затем вычислительная система начинает работу в обычном режиме. В ходе ее работы

- получают с помощью 1-го средства значения статусов name, gid, uid каждого процесса,
- получают с помощью 2-го средства значения статусов name, pid, uid каждого процесса,
- получают с помощью 3-го средства значения статусов pid, ppid, name, uid, state каждого процесса,
- получают с помощью 4-го средства значения статусов gid, name, pid, ppid, state каждого процесса.

Получение сведений осуществляется 5-м средством путем выполнения запросов ко всем остальным средствам. Все полученные сведения передаются в 5-е средство и образуют текущий список (таблицу).

После этого выполняют с помощью 5-го средства для каждого процесса из текущего списка следующие действия:

- сравнивают значения статусов name и gid, полученные из 1-го средства, с соответствующими значениями name и gid, полученными из 4-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;
- сравнивают значения статусов name и uid, полученные из 2-го средства, с соответствующими значениями name и uid, полученными из 1-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;
- сравнивают значения статусов name и pid, полученные из 2-го средства, с соответствующими значениями name и pid, полученными из 3-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;
- сравнивают значения статусов pid и uid, полученные из 3-го средства, с соответствующими значениями pid и uid, полученными из 2-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым, и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;
- сравнивают значения статусов name, pid, ppid, state, полученные из 3-го средства, с соответствующими значениями name, pid, ppid, state, полученными из 4-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов.

В результате, формируется отчет (например, в виде текстового файла), в котором будут содержаться сведения о скрытых приложениях и процессах в вычислительной системе. Этот отчет может предоставляться администратору вычислительной системы и/или пользователям по выбору.

Реализацию способа можно производить в ходе работы вычислительной системы однократно, в заданные моменты времени или, что более предпочтительно, периодически.

В общем случае, отчет может содержать перечень скрытых приложений и процессов. В этом случае администратор вычислительной системы и/или пользователь, имеющий достаточную квалификацию, будет осведомлен о наличии потенциально вредоносных программ в системе и может, при необходимости, предпринять меры по удалению (нейтрализации) этих приложений и процессов одним из известных методов. Если же отчет пустой, то работа вычислительной системы может продолжаться в обычном режиме.

(57) Формула изобретения

Способ обнаружения скрытого программного обеспечения в вычислительной системе, работающей под управлением POSIX-совместимой операционной системы, причем операционная система, помимо ядра, включает следующие программные средства:

1-е средство, выполненное с возможностью определять количество приложений, способных к запуску посредством операционной системы в вычислительной системе,

их статусы name, gid, uid, а также индексировать сопровождающие их сервисы, службы и метаданные;

2-е средство, выполненное с возможностью определять количество запущенных процессов в вычислительной системе, а также их статусы name, pid, uid;

3-е средство, выполненное с возможностью определять для запущенных процессов статусы pid, ppid, name, uid, state;

4-е средство, выполненное с возможностью определять для запущенных процессов статусы gid, name, pid, ppid, state;

5-е средство, выполненное с возможностью сравнивать результаты работы 1-го, 2-го, 3-го и 4-го средств;

способ, заключающийся в том, что

получают с помощью 1-го средства значения статусов name, gid, uid каждого процесса;

получают с помощью 2-го средства значения статусов name, pid, uid каждого процесса;

получают с помощью 3-го средства значения статусов pid, ppid, name, uid, state каждого процесса;

получают с помощью 4-го средства значения статусов gid, name, pid, ppid, state каждого процесса;

выполняют с помощью 5-го средства для каждого приложения, список которых получен с помощью 1-го средства, следующие действия:

сравнивают значения статусов name, gid, полученные из 1-го средства, с соответствующими значениями name, gid, полученными из 4-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

сравнивают значения статусов name и uid, полученные из 2-го средства, с соответствующими значениями name и uid, полученными из 1-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

сравнивают значения статусов name и pid, полученные из 2-го средства, с соответствующими значениями name и pid, полученными из 3-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

сравнивают значения статусов pid и uid, полученные из 3-го средства, с соответствующими значениями pid и uid, полученными из 2-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

сравнивают значения статусов name, pid, ppid, state, полученные из 3-го средства, с соответствующими значениями name, pid, ppid, state, полученными из 4-го средства; если хотя бы одно из значений одноименных статусов не существует или не совпадает - приложение считается скрытым и сведения о приложении заносятся в отчет о наличии скрытых приложений и процессов;

предоставляют отчет о наличии скрытых приложений и процессов;

при необходимости удаляют из вычислительной системы выявленные скрытые приложения.