(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 500 946**

(51) Int. CI.:

G06F 21/62 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **19.09.2011 E 11181801 (9)**

(97) Fecha y número de publicación de la concesión europea: **11.06.2014 EP 2431906**

(54) Título: **Procedimiento para verificar acciones con datos**

(30) Prioridad:

20.09.2010 DE 102010037651

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
01.10.2014

(73) Titular/es:

KOBIL SYSTEMS GMBH (100.0%)
Pfortenring 11
67547 Worms, DE

(72) Inventor/es:

KOYUN, ISMET

(74) Agente/Representante:

VEIGA SERRANO, Mikel

DESCRIPCIÓN

Procedimiento para verificar acciones con datos

Sector de la técnica

La presente invención se refiere a un procedimiento, que comprende verificar si una acción con datos incumple una directriz de acción; permitir la acción, en caso de que no incumpla la directriz de acción, e impedir la acción, en caso de que incumpla la directriz de acción, realizándose el procedimiento por un agente de programa, que se ejecuta en una instalación de procesamiento de datos y está asociado con los datos.

Estado de la técnica

En el estado de la técnica se conocen soluciones para evitar pérdidas de datos (*Data Loss Prevention*, DLP). Tales soluciones de DLP evitan que se extraigan datos, en particular datos confidenciales, sin autorización de la zona controlada por las soluciones de DLP, como una red fiable local como una red de empresa, por ejemplo mediante copiado en dispositivos de soporte de datos portátiles. Las soluciones de DLP conocidos en el estado de la técnica se basan en reglas basadas en usuario, es decir en función de los derechos de usuario, un usuario está autorizado o no por ejemplo a copiar datos en un dispositivo de soporte de datos portátil.

Sin embargo, una vez que los datos se encuentran fuera de la zona controlada por la solución de DLP, ya no es posible controlar la difusión de los datos. Más bien depende del cuidado de los usuarios.

Con el importante incremento de la movilidad y el aumento del número de usuarios relacionado con el mismo como por ejemplo de trabajadores de servicio externo, que tienen que llevarse los datos consigo por ejemplo también fuera de redes fiables locales como redes de empresa o acceder a los mismos, este control deficiente de la difusión de los datos se considera cada vez más crítico.

Esta problemática se agrava por la existencia cada vez mayor de programas de espionaje (por ejemplo los denominados “troyanos”), puesto que éstos permiten incluso espiar datos cifrados. Por ejemplo, pueden espiarse datos que se encuentran en un dispositivo de soporte de datos portátil protegido mediante técnica de cifrado (por ejemplo, en un lápiz de memoria USB), una vez que los datos cifrados del dispositivo de soporte de datos portátil se han puesto a disposición del sistema operativo, por ejemplo mediante la introducción de un código de descifrado, y como nueva unidad de disco se han integrado (“montado”) en la estructura de unidad de disco. En este caso tanto el usuario como un programa de espionaje puede acceder a los datos de manera transparente, es decir, sin que sea necesaria una nueva introducción del código de descifrado; los datos se descifran automáticamente a demanda (“*on-the-fly*”, sobre la marcha).

Por la solicitud de patente US 2007/214332 A1 se conoce un programa de controlador de dispositivo con una unidad de memoria para un directorio, en el que se permite un acceso, y con una unidad de decisión de permiso de acceso. La unidad de memoria para el directorio, en el que se permite un acceso, almacena como directorio, en el que se permite un acceso, un directorio de activación para un proceso, que puede acceder a una memoria S (memoria privada). La unidad de decisión de permiso de acceso verifica si un directorio de activación para un proceso, que ha solicitado acceso a la memoria S, corresponde al directorio, en el que se permite un acceso. Basándose en el resultado, la unidad de decisión de permiso de acceso decide si se aceptará la solicitud.

Por la solicitud de patente WO 2009/095413 se conoce un procedimiento y un sistema para acceder a archivos cifrados. El procedimiento presenta las etapas de: recibir una solicitud de acceso para un archivo cifrado; determinar la aplicación, que realiza la solicitud; verificar si la aplicación está autorizada para el acceso; y, en caso de que la aplicación esté autorizada, permitir el acceso.

Por la solicitud de patente DE 10 2008 028703 A1 se conoce un procedimiento para gestionar datos en un soporte de datos portátil, en el que está configurado un sistema de archivo. El procedimiento comprende la etapa de acceder a un directorio del sistema de archivo por un usuario del soporte de datos. A este respecto, al directorio está asignada al menos una operación criptográfica, de modo que al acceder al directorio se permite una realización de la operación criptográfica asignada al directorio. En este caso puede pedirse al usuario en caso de acceder a un directorio, que permite un descifrado de datos o una firma de datos por medio de una firma criptográfica, que se autentique con respecto al soporte de datos.

Por la solicitud de patente EP 1 901 193 A2 se conoce un soporte con programas de control de acceso codificados, que pueden ejecutarse, que están almacenados en el soporte de datos. En ésta se accede a los programas por un aparato con un módulo de seguridad, que se proporciona como plataforma fiable. El módulo de seguridad tiene una clave, que está asociada con los programas de control de acceso. Los programas de control de acceso controlan el acceso a datos de acceso protegido en función de reglas de acceso, que se almacenan en un archivo de configuración, estando almacenados los datos de acceso protegido en una zona de memoria de acceso protegido.

Por la solicitud de patente WO2009/158305 A1 y la publicación de GUSTAV NEUMANN ET AL: “An Approach to Engineer and Enforce Context Constraints in an RBAC Environment” (XP055067217) se conoce la consideración del entorno de un agente de programa en la determinación de una directriz de acción y la verificación de la admisibilidad de una acción.

Objeto de la invención

Por tanto, un objetivo de la presente invención es superar las desventajas mencionadas anteriormente.

Este objetivo se alcanza mediante las reivindicaciones independientes. Configuraciones a modo de ejemplo ventajosas de la invención se deducirán de las reivindicaciones dependientes.

Un primer procedimiento según la invención comprende verificar si una acción con datos incumple una directriz de acción; permitir la acción, en caso de que no incumpla la directriz de acción, e impedir la acción, en caso de que incumpla la directriz de acción, realizándose el procedimiento por un agente de programa, que se ejecuta en una instalación de procesamiento de datos y que está asociado con los datos y que en particular no está asociado con datos adicionales en la instalación de procesamiento de datos.

Una acción con datos es, por ejemplo, una acción que requiere un acceso de lectura y/o escritura a los datos. Sin embargo, también puede referirse a cualquier otra acción con datos. Por ejemplo, una acción con datos también puede ser leer y/o utilizar el mapa de memoria de datos que se encuentra en una memoria principal.

La acción con los datos puede provocarse por ejemplo mediante una entrada de usuario y/o mediante un programa informático, que se ejecuta en la instalación de procesamiento de datos. Una acción con datos es por ejemplo la parte más pequeña de una secuencia de acciones como por ejemplo el desplazamiento de datos, que puede comprender tanto el copiado de los datos como el borrado de los datos originales. Por ejemplo, es concebible que se permita una acción de una secuencia de acciones y se impida otra.

Los programas informáticos se ejecutan en particular como procesos en instalaciones de procesamiento de datos. Por proceso se entienden por ejemplo instancias de un programa informático que se ejecuta en una instalación de procesamiento de datos. Un proceso comprende por ejemplo el mapa de memoria del código de programa del programa informático en la memoria principal, una zona de memoria adicional en la memoria principal para los datos y medios operativos adicionales proporcionados por el sistema operativo o su núcleo. Por ejemplo, puede haber varios procesos de un programa informático, que según la aplicación también pueden desarrollarse al mismo tiempo o en paralelo.

El sistema operativo de una instalación de procesamiento de datos puede ser por ejemplo un sistema operativo Windows, UNIX, Linux, DOS o MAC. Un sistema operativo es un programa informático, que posibilita el uso de una instalación de procesamiento de datos. Por ejemplo, gestiona medios operativos como aparatos de almacenamiento, entrada y salida, pone a disposición funciones básicas y controla la ejecución de programas.

El núcleo de un sistema operativo forma la capa de software más inferior de una instalación de procesamiento de datos y tiene acceso directo al hardware. Pone a disposición de las aplicaciones superiores por ejemplo las siguientes funciones: interfaces con el hardware (por ejemplo aparatos de entrada/salida), gestión de memoria (por ejemplo memoria principal física y virtual), gestión de procesos, gestión de aparatos y/u organización y gestión de datos (por ejemplo sistemas de archivo, estructura de unidad de disco). Por ejemplo, puede no pertenecer al núcleo del sistema operativo una interfaz de usuario gráfica.

La directriz de acción regula por ejemplo si está permitida una acción con datos, es decir si se admite, o si está prohibida, es decir, se impide. Por ejemplo, una directriz de acción puede estar configurada en forma de matriz, asignando la matriz a cada acción con datos una regulación correspondiente.

Permitir la acción con datos significa por ejemplo que se realiza la acción con los datos; e impedir la acción con los datos significa por ejemplo que se evita la realización de la acción con los datos.

Por ejemplo instrucciones de programa, como por ejemplo llamadas de función, pueden provocar que la instalación de procesamiento de datos realice la acción con los datos. Según la invención, estas instrucciones de programa por ejemplo pueden interceptarse y/o interrumpirse y (a continuación) verificarse. En caso de que la verificación de estas instrucciones de programa no dé como resultado un incumplimiento de una directriz de acción, por ejemplo se ejecutan y/o continúan. Sin embargo, si incumplen una directriz de acción, por ejemplo se evitan y/o interrumpen y se indica al usuario que esta acción incumple una directriz de acción.

Un agente es por ejemplo un programa informático, que se ejecuta como proceso en segundo plano y que puede presentar un determinado comportamiento propio, es decir, que puede tomar decisiones de manera inadvertida y sin intervención del usuario. El agente de programa puede contener por ejemplo instrucciones de programa, que controlan el desarrollo del procedimiento, y para la verdadera realización del procedimiento acceder al menos

parcialmente a funciones, que se ponen a disposición de la instalación de procesamiento de datos por el sistema operativo y/o interfaces de programación. Por ejemplo, la acción con los datos puede realizarse con ayuda de funciones, que pone a disposición una interfaz de programación del sistema operativo. Además es concebible que el procedimiento no sólo se realice mediante el agente de programa sino que también actúen agentes y/o programas informáticos adicionales en la realización del procedimiento.

Los datos asociados con el agente de programa están almacenados por ejemplo junto con el agente de programa en un aparato electrónico, estando asociados entonces los datos en el aparato electrónico obligatoriamente (a través del lugar de almacenamiento común) con el agente de programa. El aparato electrónico es por ejemplo un dispositivo de soporte de datos, en particular un dispositivo de soporte de datos según la invención. El aparato electrónico está configurado preferiblemente como aparato de almacenamiento, que dispone de una interfaz de bus en serie universal (*Universal Serial Bus*, USB), en particular como lápiz de memoria USB o teléfono móvil, etc. Por ejemplo, el agente de programa puede verificar todas las acciones con datos, que están almacenadas en el aparato electrónico y por ello están asociadas con el agente de programa.

Los datos deben entenderse en particular como asociados con el agente de programa, cuando éste está configurado para verificar si una acción con estos datos incumple una directriz de acción.

Sin embargo, también es concebible que los datos se asocien de otro modo con el agente de programa. Por ejemplo datos, que están asociados con el agente de programa, pueden haberse registrado anteriormente en el agente de programa, o los datos asociados con el agente de programa sólo pueden descifrarse con su ayuda y por ello están asociados con el agente de programa.

Los datos asociados con el agente de programa son por ejemplo datos confidenciales, pudiendo no estar asociados datos adicionales, que no son confidenciales, por ejemplo con el agente de programa.

La instalación de procesamiento de datos es por ejemplo cualquier instalación de procesamiento de datos dentro o fuera de una red fiable local como una red de empresa, en particular una instalación de procesamiento de datos central. Una instalación de procesamiento de datos es un dispositivo que está configurado para el procesamiento de datos, en particular con ayuda de programas informáticos, por ejemplo un ordenador, un cliente ligero, un servidor y/o un ordenador portátil como un teléfono móvil, un portátil y/o un asistente personal digital (PDA).

El primer procedimiento según la invención permite verificar y dado el caso evitar acciones con datos en caso de que estas acciones por ejemplo lleven o puedan llevar a una difusión no deseada de los datos.

Un segundo procedimiento comprende la transmisión de datos desde una instalación de procesamiento de datos (por ejemplo a un aparato electrónico unido de manera separable con la instalación de procesamiento de datos, por ejemplo a un lápiz de memoria USB o un teléfono móvil, etc.) sólo en caso de que los datos transmitidos se asocien obligatoriamente con un agente de programa, realizando el agente de programa el primer procedimiento según la invención anterior cuando se ejecuta en una instalación de procesamiento de datos.

La instalación de procesamiento de datos es por ejemplo una instalación de procesamiento de datos dentro de una red fiable local como una red de empresa, en particular una instalación de procesamiento de datos propia.

Por ejemplo, es concebible que se intercepten y verifiquen instrucciones de programa como llamadas de función para la transmisión de datos desde la instalación de procesamiento de datos a un aparato electrónico. En caso de que los datos en el aparato electrónico por ejemplo se asocien obligatoriamente con el agente de programa, se permite la transmisión. En caso contrario se impide la transmisión.

Alternativamente es concebible que se provoque una asociación de los datos con el agente de programa (por ejemplo mediante un registro) y sólo en caso de que esto no sea posible, se impide la transmisión (por ejemplo el copiado de datos). Por transmitir puede entenderse por ejemplo también el copiado de los datos en una zona de memoria (por ejemplo en un fichero de archivo ejecutable), que en una emulación de un aparato electrónico mediante una instalación de procesamiento de datos se integra como unidad de disco virtual en la estructura de unidad de disco de la instalación de procesamiento de datos (que realiza la emulación).

Mediante el segundo procedimiento se evita en particular que puedan extraerse datos de una red fiable local, sin que se garantice una protección mediante el procedimiento según la invención. Transmitir significa entre otras cosas copiar, mover o extraer.

El aparato electrónico es, como se explicó anteriormente, por ejemplo un dispositivo de soporte de datos, en particular un dispositivo de soporte de datos según la invención.

Por que puede unirse de manera separable se entenderá por ejemplo que el aparato electrónico está unido con una interfaz de datos de la instalación de procesamiento de datos y puede leerse y/o describirse por ésta. A este respecto, la unión puede ser tanto lógica como física (mecánica). Por ejemplo es concebible una unión por cable, por

ejemplo a través de una interfaz de datos en serie (por ejemplo una interfaz USB, *Firewire*, RS-232, etc.) o una interfaz de datos en paralelo (por ejemplo una interfaz de sistema para ordenadores pequeños (*Small Computer sistema Interface*) SCSI, IEEE-1284, etc.). La unión por cable puede producirse de manera mecánica, por ejemplo mediante inserción y en particular es reversible. Por otro lado también es concebible una unión inalámbrica, por ejemplo a través de una interfaz de datos por radio (por ejemplo una interfaz de red inalámbrica de área local (*Wireless Local Area Network*) WLAN, *Bluetooth*, etc.) o una interfaz de datos por infrarrojos (por ejemplo una interfaz de asociación de datos por infrarrojos (*Infrared Data Association*) IrDA, etc.). En este caso no es necesaria una unión mecánica entre el aparato electrónico y la instalación de procesamiento de datos.

El programa informático según la invención comprende instrucciones de programa, en el que las instrucciones de programa provocan que un procesador realice el procedimiento según la invención, cuando el programa informático se ejecuta mediante el procesador. A este respecto, el procesador por ejemplo puede formar parte de la instalación de procesamiento de datos, en la que se realiza el procedimiento correspondiente.

El programa informático según la invención puede estar configurado por ejemplo como agente. Por ejemplo, el programa informático según la invención, que comprende instrucciones de programa para la realización del primer procedimiento según la invención, puede ser el agente de programa. Un programa informático, que provoca que un procesador realice el segundo procedimiento, puede estar configurado por ejemplo como agente de programa local, que preferiblemente se ejecuta de manera permanente en una instalación de procesamiento de datos. Por consiguiente debe distinguirse entre el agente de programa y el agente de programa local. El programa informático se carga por ejemplo al iniciar la instalación de procesamiento de datos y después por ejemplo se ejecuta sin interrupción en la misma.

El programa informático según la invención puede estar configurado al menos parcialmente como biblioteca dinámica (biblioteca de enlace dinámico, *Dynamic Link Library*, DLL).

Un programa informático puede difundirse por ejemplo a través de una red como Internet, una red telefónica o de radiotelefonía móvil y/o una red local. Un programa informático puede ser al menos parcialmente un software y/o *firmware* de un procesador.

El dispositivo de soporte de datos según la invención puede ser por ejemplo un medio de almacenamiento legible por ordenador, que presenta un programa informático según la invención y por ejemplo está configurado como medio de almacenamiento magnético, eléctrico, electromagnético, óptico y/o de otro tipo.

Un dispositivo de soporte de datos de este tipo es por ejemplo portátil o está instalado de manera fija en un dispositivo. Ejemplos de un dispositivo de soporte de datos de este tipo son memorias no volátiles con acceso aleatorio (RAM) como por ejemplo memorias flash NOR o con acceso secuencial como memorias flash NAND y/o memorias con acceso de sólo lectura (ROM) o acceso de escritura-lectura. Legible por ordenador se entenderá en particular como que el medio de almacenamiento puede leerse y/o describirse por un ordenador o una instalación de procesamiento de datos, por ejemplo por un procesador.

Instalaciones de procesamiento de datos (por ejemplo la instalación de procesamiento de datos central y la propia) comprenden medios para la realización de al menos uno del primer procedimiento según la invención y el segundo procedimiento, estando configuradas las instalaciones de procesamiento de datos en relación con el software para poder realizar el procedimiento.

A este respecto, por configurado en relación con el software se entenderá en particular la preparación de la instalación de procesamiento de datos que es necesaria para poder realizar un procedimiento por ejemplo en forma de un programa informático. Esta preparación se denomina a menudo instalación.

Un controlador es un programa informático, que en particular pone a disposición del sistema operativo funciones para controlar un componente de hardware. Los controladores instalados se cargan por ejemplo al iniciar la instalación de procesamiento de datos junto con el sistema operativo y por ejemplo forman parte del núcleo del sistema operativo, que se ejecuta como proceso en la instalación de procesamiento de datos.

Por ejemplo, un controlador de un dispositivo de soporte de datos pone a disposición del sistema operativo datos, que están almacenados en la memoria del dispositivo de soporte de datos, tras su instalación, de modo que el sistema operativo puede integrar ("montar") la memoria del dispositivo de soporte de datos por ejemplo como unidad de disco nueva en la estructura de unidad de disco de la instalación de procesamiento de datos, una vez que se una el dispositivo de soporte de datos con la instalación de procesamiento de datos.

Tanto en la instalación de procesamiento de datos central como en la propia puede estar instalado por ejemplo un controlador para el dispositivo de soporte de datos según la invención, que pone a disposición del sistema operativo los datos contenidos en el mismo. Además es concebible que en algunas o en todas las instalaciones de procesamiento de datos dentro de la red fiable local (las instalaciones de procesamiento de datos propias) esté

instalado el agente de programa local. El agente de programa local se carga por ejemplo al iniciar las instalaciones de procesamiento de datos y se ejecuta sin interrupción en las mismas.

Es concebible, que el programa informático según la invención se ejecute con medios de la instalación de procesamiento de datos según la invención, por ejemplo un procesador. Por procesador se entenderán, entre otros, unidades de control, microprocesadores, microunidades de control como microcontroladores, procesadores digitales de señales (DSP), circuitos integrados de aplicación específica (ASIC) o disposiciones de puertas programables en campo (FPGA).

Las instalaciones de procesamiento de datos comprenden por lo demás medios para su unión con el dispositivo de soporte de datos según la invención. Estos medios pueden ser por ejemplo una interfaz de datos correspondiente a la interfaz de datos del dispositivo de soporte de datos.

Un sistema comprende un dispositivo de soporte de datos según la invención y una instalación de procesamiento de datos, que puede unirse de manera separable con el dispositivo de soporte de datos. Mediante un sistema de este tipo pueden verificarse y dado el caso evitarse acciones con datos también fuera de una red fiable local como una red de empresa, para por ejemplo evitar una difusión y/o uso no deseado de los datos.

A continuación se describen configuraciones a modo de ejemplo de la presente invención, que se refieren a características a modo de ejemplo adicionales del procedimiento según la invención, del programa informático según la invención, de las instalaciones de procesamiento de datos así como del dispositivo de soporte de datos según la invención y del sistema. En particular mediante la descripción de una etapa de procedimiento adicional de uno de los procedimientos también se darán a conocer medios para la realización de la etapa de procedimiento de la instalación de procesamiento de datos o del dispositivo de soporte de datos correspondiente y una instrucción de programa correspondiente del programa informático correspondiente, que provoca que un procesador realice la etapa de procedimiento cuando el programa informático se ejecuta mediante el procesador. Lo mismo será válido también para dar a conocer un medio para la realización de una etapa de procedimiento o una instrucción de programa, por ejemplo al dar a conocer un medio para la realización de una etapa de procedimiento también se dará a conocer la etapa de procedimiento correspondiente y una instrucción de programa correspondiente.

En configuraciones a modo de ejemplo de la invención la acción con los datos parte de uno o varios programas, que se ejecutan en la instalación de procesamiento de datos. Es decir, se trata de una acción con datos, que parte de una instancia de la instalación de procesamiento de datos. Esta acción puede provocarse por ejemplo mediante un usuario de la instalación de procesamiento de datos y/o un programa informático como un programa de espionaje, que se ejecuta en la instalación de procesamiento de datos. A diferencia de un cortafuegos o un programa cortafuegos, según la invención por ejemplo también se verifican y dado el caso se impiden intentos de acceso internos a los datos asociados con el agente de programa.

En configuraciones a modo de ejemplo de la invención la acción con los datos está configurada como llamada de programa, procedimiento y/o interfaz, a la que como argumento se transfieren datos asociados con el agente de programa, y/o como intento de acceso a datos asociados con el agente de programa.

A este respecto, por procedimiento se entienden por ejemplo instrucciones de programa secuenciales que permiten realizar una determinada tarea de manera repetida. Un procedimiento es por ejemplo una función, un subprograma, un método y/o una subrutina.

Llamadas de interfaz son por ejemplo llamadas de una función de una interfaz de programación, que por ejemplo se ponen a disposición mediante el sistema operativo de la instalación de procesamiento de datos o su núcleo. Por ejemplo los sistemas operativos Windows, UNIX, Linux, DOS o MAC pueden poner a disposición tales interfaces de programación.

Las interfaces y/o interfaces de programación están configuradas por ejemplo como controladores y/o como bibliotecas dinámicas. En un sistema operativo Windows entran por ejemplo la interfaz de programación de aplicación de Windows (*Windows Application Programming Interface*, WINAPI) y la interfaz de programación de zócalo de Windows (*Windows Socket Application Programming Interface*, Winsock-API). En un sistema operativo Linux una interfaz de programación de este tipo es por ejemplo la denominada interfaz de programación de núcleo Linux (Linux-Kernel API). En un sistema operativo MAC se encuentra por ejemplo la interfaz de programación de Cocoa, Carbon y POSIX. Un procedimiento y/o una función de una interfaz de programación puede llamarse por ejemplo por un programa, que se ejecuta en la instalación de procesamiento de datos.

Por ejemplo es concebible, que el agente de programa, una vez cargado y una vez se ejecute en la instalación de procesamiento de datos, esté interconectado en la comunicación (interna) del procesamiento de datos, por ejemplo de tal manera que desvíe la comunicación (interna) de la instalación de procesamiento de datos o partes de la comunicación (interna) de la instalación de procesamiento de datos a sí mismo y por ejemplo sólo la siga transmitiendo tras una verificación positiva. Por ejemplo, el agente de programa se engancha antes o después a la

comunicación entre procesos y decide si se realiza esta transmisión o el proceso (o la función), que deberá obtener los datos.

La comunicación interna de una instalación de procesamiento de datos puede referirse por ejemplo a la comunicación entre procesos, que se ejecutan en la instalación de procesamiento de datos, con otros procesos, programas y/o al sistema operativo de la instalación de procesamiento de datos; a éstos pertenecen por ejemplo llamadas de programa, procedimiento y/o interfaz. Sin embargo, la comunicación interna de una instalación de procesamiento de datos también puede referirse por ejemplo a la comunicación entre diferentes unidades físicas de la instalación de procesamiento de datos, como memoria principal, memoria de programa, procesador y/o interfaz de datos, etc.

Por ejemplo se desvían llamadas de programa, procedimiento y/o interfaz generales al agente de programa. Esta desviación puede producirse por ejemplo por medio de "API-*Hooking*" (enlace API). Cuando una acción con datos representa una llamada de una función desviada de este tipo de una interfaz de programación, el agente de programa puede verificar la acción con los datos.

Por ejemplo, es concebible que el sistema operativo tenga una tabla de llamada de sistema (*System Call Table*), en la que se enlazan llamadas de funciones de una interfaz de programación con la función correspondiente del sistema operativo. Si se modifica un enlace de este tipo, de modo que remite al agente de programa en lugar de a la función correspondiente del sistema operativo, entonces se desvía una llamada correspondiente al agente de programa. En principio, el "enlace API" es concebible en todos los sistemas operativos que ponen a disposición interfaces de programación. Pueden ser por ejemplo sistemas operativos Windows, UNIX, Linux, DOS o MAC.

La transferencia de datos asociados con el agente de programa como argumento se produce por ejemplo por medio de un parámetro de referencia (por ejemplo de un indicador de los datos o de una ruta) y/o de un parámetro de valor (por ejemplo los propios datos).

En configuraciones a modo de ejemplo de la invención, el agente de programa está almacenado en un aparato electrónico, que puede unirse de manera separable con la instalación de procesamiento de datos. El aparato electrónico es por ejemplo un dispositivo de soporte de datos, en particular un dispositivo de soporte de datos según la invención.

Como se explicó anteriormente, por que puede unirse de manera separable se entenderá que el aparato electrónico puede unirse con una interfaz de datos de la instalación de procesamiento de datos y a continuación leerse por la misma. El aparato electrónico está configurado preferiblemente como aparato de almacenamiento USB, en particular como lápiz de memoria USB.

Según la invención el agente de programa puede ejecutarse directamente desde el aparato electrónico en la instalación de procesamiento de datos.

Por ejemplo el agente de programa puede cargarse directamente (es decir, inmediatamente) desde una memoria del aparato electrónico en una memoria principal de un procesador de la instalación de procesamiento de datos, para ejecutarse por la misma. Así, el agente de programa, aunque se ejecute en la instalación de procesamiento de datos, no deja huellas restantes en la misma, lo que dificulta a los atacantes el análisis y/o la manipulación del agente de programa. Además es concebible, que el agente de programa garantice que tras su finalización no permanezca en la instalación de procesamiento de datos, borrando o haciendo que se borren por ejemplo las zonas correspondientes en la memoria principal virtual y/o física de la instalación de procesamiento de datos.

En caso de que el agente de programa no permanezca en la instalación de procesamiento de datos, también en el caso de cada nueva unión del aparato electrónico con la instalación de procesamiento de datos se ejecuta directamente desde el aparato electrónico en la instalación de procesamiento de datos. Esto significa por ejemplo, que el agente de programa tiene que ejecutarse cada vez directamente desde el aparato electrónico en la instalación de procesamiento de datos.

Por ejemplo es concebible, que el agente de programa se cargue en la memoria principal del procesador, antes de que pueda realizarse una acción con los datos, en particular una vez que el aparato electrónico está unido con la instalación de procesamiento de datos. De este modo se evita que se realice una acción con los datos, antes de que el agente de programa pueda verificarla para determinar un incumplimiento de una directriz de uso.

Según la invención, los datos asociados con el agente de programa están almacenados al menos temporalmente en el aparato electrónico. Según el segundo procedimiento, los datos se transmiten al aparato electrónico, que está unido de manera separable con la instalación de procesamiento de datos y en el que por ejemplo también está almacenado el agente de programa.

El aparato electrónico puede comprender por ejemplo dos memorias, conteniendo la primera memoria el agente de programa y la segunda memoria los datos asociados con el agente de programa. Esto es en particular ventajoso

cuando con la unión con la instalación de procesamiento de datos inicialmente sólo se pone a disposición del sistema operativo la primera memoria, que contiene el agente de programa.

Las memorias pueden estar configuradas por ejemplo de diferente manera. En particular la primera memoria, que contiene el agente de programa, puede estar configurada como memoria de sólo lectura, para evitar una manipulación del agente de programa. Por el contrario, la segunda memoria está configurada por ejemplo como memoria de escritura-lectura no volátil.

Por ejemplo es concebible, que ambas memorias sean sólo memorias virtuales y pertenezcan a la misma memoria física, que sin embargo está subdividida en varias zonas de memoria (por ejemplo particiones), correspondiendo cada zona de memoria por ejemplo a una de las memorias virtuales.

También son concebibles configuraciones a modo de ejemplo, en las que el aparato electrónico dispone de más de dos memorias. Por ejemplo, una o varias memorias adicionales pueden servir como memoria intermedia para el cifrado y/o descifrado y/o para la recepción y/o el envío de datos.

En configuraciones a modo de ejemplo de la invención, los datos asociados con el agente de programa están almacenados de manera cifrada en el aparato electrónico. El descifrado de los datos asociados con el agente de programa puede ser posible por ejemplo sólo con ayuda de un código de descifrado. El código de descifrado puede estar almacenado por ejemplo en una tarjeta inteligente y/o introducirse por el usuario. A partir del código de descifrado puede generarse por ejemplo la clave para el descifrado de los datos.

Además la clave puede generarse por ejemplo al menos parcialmente a partir de información biométrica (por ejemplo una huella digital y/o huella del iris) de un usuario.

Mediante el cifrado se evita que una persona obtenga acceso a los datos, cuando está en posesión del aparato electrónico, en el que se encuentran los datos, y/o en posesión de una copia de los datos, no sin embargo en posesión del código de descifrado.

Además es concebible, que el código de descifrado tenga que combinarse con una propiedad del aparato electrónico como por ejemplo un número de serie, una dirección MAC y/o una designación de ruta, por ejemplo por medio de un enlace XOR, para generar la clave para el descifrado. De este modo es posible el descifrado de los datos sólo en relación con el aparato electrónico. No puede descifrarse una copia de los datos, incluso cuando se conoce el código de descifrado. Un cifrado de este tipo también se denomina cifrado local.

Por ejemplo es concebible, que para el descifrado de los datos sea necesaria la ejecución de un determinado programa de descifrado. Un programa informático de este tipo podría garantizar, por ejemplo, que el agente de programa está cargado y se ejecuta en la instalación de procesamiento de datos, antes de que se descifren los datos.

El programa de descifrado está almacenado por ejemplo en el aparato electrónico y puede ejecutarse directamente desde el mismo en la instalación de procesamiento de datos. Por ejemplo forma parte del agente de programa y/o está configurado como agente de descifrado separado. Alternativamente, el programa de descifrado está configurado por ejemplo como controlador para el aparato electrónico, que debe instalarse en la instalación de procesamiento de datos.

El programa de descifrado y/o el agente de programa están contenidos por ejemplo en una primera memoria no cifrada del aparato electrónico, mientras que los datos asociados con el agente de programa están contenidos en una segunda memoria cifrada del aparato electrónico.

Por ejemplo pueden ponerse a disposición del sistema operativo datos cifrados mediante la generación de la clave y la ejecución del programa de descifrado de tal manera que es posible un acceso transparente a los datos, es decir los datos se descifran automáticamente a demanda (“sobre la marcha”) mediante el programa de descifrado y no es necesaria una nueva introducción del código de descifrado. En este caso, durante el acceso a los datos no puede observarse ninguna diferencia con respecto a los datos no cifrados, una vez que se ha introducido el código de descifrado correcto. Alternativamente, por ejemplo el programa de descifrado, antes de cada descifrado de datos cifrados, puede hacer necesaria de nuevo la introducción de un código de descifrado, que por ejemplo es diferente para cada conjunto de datos (por ejemplo cada archivo).

Además el descifrado depende, por ejemplo, de la verificación de la acción con los datos, siendo los datos los datos que van a descifrarse y estando asociados con el agente de programa.

Los datos descifrados se almacenan por ejemplo de manera intermedia en una memoria intermedia del aparato electrónico y/o se asocian con el agente de programa para evitar que permanezcan en la instalación de procesamiento de datos y/o puedan espiarse por programas como por ejemplo programas de espionaje, que se ejecutan en la instalación de procesamiento de datos.

En este contexto en particular también es concebible que se evite una transferencia de los archivos descifrados desde la memoria principal física de la instalación de procesamiento de datos a una memoria principal virtual (por ejemplo un archivo de transferencia), el denominado "*swap*", y/o se borren los archivos descifrados en la misma antes de finalizar el agente de programa.

En configuraciones a modo de ejemplo de la invención, los datos asociados con el agente de programa sólo se descifran en caso de que la acción no incumpla ninguna directriz de acción. A este respecto, los datos asociados con el agente de programa pueden estar cifrados por ejemplo mientras no se utilicen. Esto posibilita descifrar los datos sólo cuando con ellos se realizan realmente acciones. Así un programa de espionaje que se ejecuta en la instalación de procesamiento de datos, en todo caso, puede espiar sólo los datos con los que se realizan acciones. Todos los demás datos permanecen cifrados. Los datos descifrados se asocian por ejemplo también con el agente de programa.

En configuraciones a modo de ejemplo de la invención, los datos asociados con el agente de programa pueden cargarse a través de una interfaz inalámbrica del aparato electrónico en una memoria del aparato electrónico y/o de la instalación de procesamiento de datos. Una interfaz inalámbrica es por ejemplo una interfaz de datos por radio (por ejemplo *Wireless Local Area Network* WLAN, *Bluetooth*) y/o una interfaz de datos por infrarrojos (por ejemplo IrDA), etc.

Por ejemplo, el aparato electrónico puede presentar medios para la comunicación por radio, en particular para la comunicación por radiotelefonía móvil (preferiblemente a través de una norma del Proyecto de Asociación de 3ª Generación como la norma de sistema global para comunicaciones móviles GSM (*Global System for Mobile Communications*), servicio general de radio por paquetes GPRS (*General Packet Radio Service*) y/o acceso de paquetes de enlace descendente de alta velocidad HSDPA (*High Speed Downlink Packet Access*) o una evolución de las mismas), para cargar por ejemplo los datos, que están almacenados en un servidor, en una memoria propia (por ejemplo una memoria intermedia) y/o una memoria de la instalación de procesamiento de datos. Estos datos se asocian por ejemplo con el agente de programa, una vez se cargan en la memoria propia y/o la memoria de la instalación de procesamiento de datos.

De este modo se posibilita el acceso móvil a datos, por ejemplo datos confidenciales, que están almacenados dentro de una red fiable local (por ejemplo en un servidor), desde fuera, sin que se pierda el control sobre el uso y/o la difusión de los datos.

Por lo demás puede determinarse por ejemplo la forma de la interfaz inalámbrica, a través de la que pueden cargarse los datos asociados con el agente de programa en una memoria del aparato electrónico y/o de la instalación de procesamiento de datos, mediante la directriz de acción. De este modo, por ejemplo fuera de una red local no fiable puede garantizarse el uso único de una transmisión por radio cifrada (segura) (por ejemplo acceso inalámbrico protegido (*Wifi Protected Access*), WPA o WPA2 y/o UMTS, aunque en determinadas circunstancias ningún GSM o privacidad equivalente por cable (*Wired Equivalent Privacy*, WEP)).

En configuraciones a modo de ejemplo de la invención, el aparato electrónico presenta medios para el cifrado y/o para el descifrado de los datos asociados con el agente de programa. Es decir, los datos asociados con el agente de programa dentro del aparato electrónico pueden cifrarse y/o descifrarse por ejemplo por medio de una unidad de criptografía. De este modo, por ejemplo, todo el descifrado de los datos tiene lugar dentro del aparato electrónico, de modo que se excluye una manipulación o un espionaje del cifrado y/o descifrado.

Es cierto que además puede ser necesaria la ejecución de un programa de descifrado mediante un procesador de la instalación de procesamiento de datos y la introducción de un código de descifrado. Sin embargo, de este modo por ejemplo sólo se libera (en lugar de generarse) la verdadera clave para el descifrado de los datos, que se almacena en la unidad de criptografía, de modo que el código de descifrado no permite deducir la clave. Alternativamente, el aparato electrónico puede presentar en sí mismo medios para la introducción del código de descifrado, por ejemplo un teclado, un teclado numérico, un escáner de huella digital y/o iris.

En configuraciones a modo de ejemplo de la invención, el aparato electrónico presenta una interfaz USB para la unión con la instalación de procesamiento de datos.

En configuraciones a modo de ejemplo de la invención, el agente de programa puede ejecutarse en la instalación de procesamiento de datos sin necesidad de instalación (es decir, por ejemplo, sin haberse realizado anteriormente una instalación de software en la instalación de procesamiento de datos).

Por ejemplo, sin necesidad de instalación puede significar que la instalación de procesamiento de datos no requiere un equipo especial en relación con el software, para poder realizar el procedimiento por ejemplo en forma de un programa informático.

Por ejemplo, el agente de programa podría ponerse a disposición del sistema operativo de la instalación de procesamiento de datos mediante un controlador previsto en relación con la norma por el sistema operativo. Sin embargo, éste no tiene que ser el caso, también son concebibles formas de realización sin un controlador normalizado de este tipo, en las que el agente de programa también puede ejecutarse sin necesidad de instalación en la instalación de procesamiento de datos.

Esto es en particular ventajoso, cuando el agente de programa está almacenado en un aparato electrónico portátil, porque en consecuencia éste puede unirse con cualquier instalación de procesamiento de datos y es posible un acceso a los datos contenidos en la misma por medio del agente de programa, sin que la instalación de procesamiento de datos requiera de algún modo un equipo especial en relación con el software, por ejemplo un equipo adaptado al agente de programa. En este caso, el agente de programa podría ejecutarse por ejemplo cada vez directamente (y sin necesidad de instalación) desde el aparato electrónico en la instalación de procesamiento de datos, cuando el aparato electrónico se une con la instalación de procesamiento de datos.

En configuraciones a modo de ejemplo de la invención, una acción con los datos sólo es posible cuando el agente de programa se ejecuta en la instalación de procesamiento de datos. Esto se consigue por ejemplo porque el agente de programa pone a disposición del sistema operativo y/o descifra los datos asociados con el mismo.

Según la invención, el primer procedimiento comprende por lo demás determinar la directriz de acción. La determinación de la directriz de acción comprende por ejemplo seleccionar una directriz de acción a partir de varias directrices de acción establecidas y se realiza preferiblemente por el agente de programa y/o un agente de perfil. Por ejemplo es concebible, que el agente de programa llame al agente de perfil para determinar la directriz de acción y que el agente de perfil tras la determinación de la directriz de acción la devuelva al agente de programa.

El agente de perfil está almacenado por ejemplo junto con el agente de programa en el aparato electrónico y se ejecuta en la instalación de procesamiento de datos.

Según la invención la determinación de la directriz de acción se basa al menos parcialmente en el entorno, en el que se ejecuta el agente de programa.

Por ejemplo, el agente de perfil compara el entorno de la instalación de procesamiento de datos con diferentes perfiles de entorno establecidos, estando asignada a cada perfil de entorno una directriz de acción. El agente de perfil selecciona la directriz de acción, que está asignada al perfil de entorno con la mayor coincidencia y/o la coincidencia exacta con el entorno de la instalación de procesamiento de datos. En caso de que no exista ninguna coincidencia y/o una coincidencia demasiado baja con los perfiles de entorno establecidos, es concebible que se seleccione una directriz de acción con autorizaciones mínimas (por ejemplo sólo lectura de datos), una directriz de acción mínima, mediante el agente de perfil. Además, la directriz de acción podría determinarse en función del usuario.

En configuraciones a modo de ejemplo de la invención, el primer procedimiento comprende por lo demás la comunicación de información sobre el entorno, en el que se ejecuta el agente de programa, a un servidor; y la recepción de información sobre la directriz de acción por el servidor, pudiendo determinarse la directriz de acción mediante el servidor basándose al menos parcialmente en la información comunicada. Por ejemplo, el agente de perfil puede ejecutarse al menos parcialmente en el servidor.

Según la invención el entorno, en el que se ejecuta el agente de programa, comprende la ubicación de la instalación de procesamiento de datos, su conexión de red y/o las aplicaciones y/o procesos instalados y/o que se ejecutan en la misma.

En caso de que la instalación de procesamiento de datos se encuentre por ejemplo dentro de una red fiable local como una red de empresa, entonces por ejemplo se selecciona una directriz de acción máxima (véase más abajo). Éste es el caso en particular cuando el agente de programa se ejecuta en una instalación de procesamiento de datos propia. Sin embargo, en caso de que la instalación de procesamiento de datos no se encuentre en ninguna red fiable local, entonces se selecciona por ejemplo la directriz de acción mínima. Éste es el caso en particular, cuando el agente de programa se ejecuta en una instalación de procesamiento de datos central.

La directriz de acción máxima permite por ejemplo casi todas las acciones con datos, como copiar, leer, modificar, imprimir, borrar, etc. La directriz de acción mínima evita por el contrario casi todas las acciones con datos, como copiar, modificar, imprimir, etc., la única acción permitida puede ser por ejemplo mostrar los datos en la pantalla.

Una directriz de acción media podría permitir por el contrario para algunos datos casi todas las acciones, para otros datos (por ejemplo especialmente datos fiables) podría permitir por el contrario sólo la lectura (es decir, los datos se clasificarían por ejemplo según la fiabilidad de manera diferente). La directriz de acción media podría seleccionarse por ejemplo cuando el agente de programa se ejecuta en una instalación de procesamiento de datos central, que está unida con la red fiable local, por ejemplo a través de una red privada virtual (VPN).

En configuraciones a modo de ejemplo de la invención, la determinación de la directriz de acción depende al menos parcialmente de una indicación de usuario.

Por ejemplo, el agente de programa y/o el agente de perfil puede informar al usuario sobre la acción con los datos y pedirle que indique si debe permitirse o impedirse esta acción. Esta información por ejemplo puede almacenarse y reutilizarse. Para evitar que programas de espionaje manipulen la entrada de usuario, ésta puede protegerse por ejemplo mediante la introducción adicional de una contraseña.

En configuraciones a modo de ejemplo de la invención, el primer procedimiento comprende por lo demás la introducción de algunos o todos los programas o procesos, que se ejecutan en la instalación de procesamiento de datos y que realizan al menos una acción con los datos, en una lista de usuario. Una vez finalizada la ejecución de los programas o procesos, éstos se borran de la lista de usuario.

Por ejemplo los programas, que realizan al menos una acción con los datos, pueden aislarse de otros programas. Por ejemplo, en principio podría impedirse su comunicación con otros programas y/o procesos y/o verificarse según la invención. También es concebible, que la comunicación con algunos programas y/o procesos se impida o verifique según la invención, aunque se permita con otros, es decir los datos se clasificarían por ejemplo según fiabilidad de manera diferente. Esto se simplifica mediante la lista de usuario.

En configuraciones a modo de ejemplo de la invención, el primer procedimiento comprende por lo demás la finalización de todos los programas o procesos que se ejecutan en la instalación de procesamiento de datos, que se han introducido en la lista de usuario, antes de finalizar el agente de programa. De este modo se evita que los programas, que realizan acciones con los datos, sigan ejecutándose en la instalación de procesamiento de datos, cuando el agente de programa ha finalizado.

Esto es importante para evitar que tras la finalización del agente de programa puedan realizarse acciones con los datos, que incumplen la directriz de acción, puesto que pueden estar presentes al menos partes de los datos por ejemplo todavía como mapas de memoria en la memoria principal de la instalación de procesamiento de datos. Estos mapas de memoria se borran por ejemplo con la finalización de los programas o procesos. Esto puede afectar por ejemplo también a archivos transferidos a una memoria principal virtual.

Por lo demás, con la finalización del agente de programa podría garantizarse que ni el agente de programa ni partes de su código de programa permanecen en la instalación de procesamiento de datos.

En configuraciones a modo de ejemplo de la invención, la directriz de acción comprende una denominada lista “negra” con programas, que no pueden realizar y/o llamar ninguna acción con datos. Esto significa, por ejemplo, que las llamadas de funciones, que realizan acciones con datos, que se asocian con el agente de programa, incumplen la directriz de acción, en caso de que partan de programas en la lista negra.

Por ejemplo, los programas en la lista negra están predefinidos, pueden establecerse y/o modificarse mediante entradas de usuario, y/o pueden determinarse y/o actualizarse debido a reglas predefinidas.

En configuraciones a modo de ejemplo de la invención, la directriz de acción comprende una denominada lista “blanca” con programas que pueden realizar y/o llamar acciones con datos. Esto significa, por ejemplo, que las llamadas de funciones, que realizan acciones con datos, que se asocian con el agente de programa, no incumplen la directriz de acción, en caso de que partan de programas en la lista blanca.

Por ejemplo, los programas en la lista blanca están predefinidos, pueden establecerse y/o modificarse mediante entradas de usuario, y/o pueden determinarse y/o actualizarse debido a reglas predefinidas.

En caso de que la directriz de acción comprenda por ejemplo sólo una lista blanca, entonces el agente de programa está configurado preferiblemente de tal manera que las acciones con los datos, que se realizan y/o llaman por otros programas (distintos de los programas introducidos en la lista blanca), o bien incumplen la directriz de acción y/o bien se pide al usuario que indique si incumplen la directriz de acción.

En configuraciones a modo de ejemplo de la invención, al menos uno de los procedimientos comprende la actualización del agente de programa y/o de agentes adicionales.

La actualización del agente de programa y/o de agentes adicionales puede realizarse en particular por un agente de actualización, que por ejemplo está almacenado junto con el agente de programa en el aparato electrónico y se ejecuta en la instalación de procesamiento de datos.

El agente de actualización puede llamar y realizar por ejemplo a intervalos regulares instrucciones de actualización, que por ejemplo deben disponer de un certificado de seguridad válido, desde un servidor. De este modo es posible una corrección de errores y/o una adaptación del agente de programa y/o de los agentes adicionales. Por ejemplo

puede producirse una adaptación a una nueva situación de amenaza, directrices internas de la empresa y/o nuevos conocimientos.

Por ejemplo es concebible, que las directrices de acción como la lista negra y/o la blanca se actualicen por el agente de actualización a intervalos regulares, para por ejemplo considerar nuevos programas de espionaje.

En configuraciones a modo de ejemplo de la invención, al menos uno de los procedimientos comprende la creación de una copia de seguridad de los datos asociados con el agente de programa, pudiendo la creación de la copia de seguridad ser dependiente al menos parcialmente de la directriz/las directrices de acción.

La creación de una copia de seguridad puede realizarse por ejemplo por un agente de copia de seguridad, que está almacenado por ejemplo junto con el agente de programa en el aparato electrónico y por ejemplo se ejecuta en la instalación de procesamiento de datos.

En configuraciones a modo de ejemplo de la invención, al menos uno de los procedimientos comprende registrar información y comunicar la información a un servidor. Preferiblemente toda la información sobre la acción con los datos debe registrarse, es decir, protocolizarse, a ésta pertenecen por ejemplo la hora, la designación de la acción (por ejemplo nombre de función), datos en cuestión (por ejemplo ruta) e instalación de procesamiento de datos operativa y/o resultado de verificación (incumplimiento/no incumplimiento).

Esta información puede enviarse por ejemplo para su almacenamiento y/o evaluación a un servidor.

El registro y la comunicación de los datos puede realizarse en particular por un agente de protocolización, que por ejemplo está almacenado junto con el agente de programa en el aparato electrónico y se ejecuta en la instalación de procesamiento de datos.

En configuraciones a modo de ejemplo de la invención, el segundo procedimiento se realiza por el agente de programa local, que se ejecuta en la instalación de procesamiento de datos.

En configuraciones a modo de ejemplo de la invención, el sistema comprende por lo demás un servidor, estando configurado el servidor para recibir, procesar y/o enviar información. Puede ser por ejemplo información sobre la acción con los datos.

En configuraciones a modo de ejemplo de la invención, el servidor está configurado para almacenar y evaluar información sobre la acción con los datos. El almacenamiento y la evaluación de la información sobre la acción con los datos puede realizarse en particular por un agente de base de datos en el servidor.

En configuraciones a modo de ejemplo de la invención, el servidor está configurado para gestionar modificaciones de los datos asociados con el agente de programa. La gestión de las modificaciones de los datos asociados con el agente de programa puede referirse por ejemplo a datos en diferentes versiones en diferentes instalaciones de procesamiento de datos y/o aparatos electrónicos. Por ejemplo, puede almacenarse una modificación de datos con un nuevo número de versión, de modo que puede realizarse un seguimiento de modificaciones de los datos y no se borran los datos originales. Para el usuario son visibles por ejemplo sólo datos con el número de versión más reciente en relación con la norma y/o en función de los derechos de usuario. La gestión de las modificaciones puede realizarse en particular por un agente de versión, que se ejecuta en el servidor.

En configuraciones a modo de ejemplo de la invención, el servidor está configurado para gestionar los aparatos electrónicos y su asignación a usuarios. Por ejemplo, el servidor puede provocar que un aparato electrónico borre los datos almacenados en el mismo, cuando se ha dado aviso de robo del aparato electrónico. La asignación y gestión de los usuarios puede realizarse en particular por un agente de gestión, que se ejecuta en el servidor.

En configuraciones a modo de ejemplo de la invención, el servidor está configurado para gestionar y restablecer códigos de descifrado para el descifrado de los datos cifrados en los aparatos electrónicos. La gestión y el restablecimiento pueden realizarse en particular por un agente de gestión de código de descifrado, que se ejecuta en el servidor.

Las configuraciones a modo de ejemplo de la presente invención, descritas en esta solicitud, también se darán a conocer en todas las combinaciones entre sí.

A partir de la siguiente descripción detallada de algunas formas de realización a modo de ejemplo de la presente invención, en particular en relación con las figuras, se deducirán configuraciones a modo de ejemplo adicionales ventajosas de la invención.

Sin embargo, las figuras adjuntas a la solicitud sólo servirán para aclarar, no sin embargo para determinar el alcance de protección de la invención. Los dibujos adjuntos no son a escala y sólo representarán el concepto general de la

presente invención a modo de ejemplo. En particular las características, que están contenidas en las figuras, no deberán considerarse de ningún modo como componente necesario de la presente invención.

Descripción de las figuras

En las figuras muestran

la figura 1: un diagrama de bloques de una forma de realización a modo de ejemplo de un dispositivo de soporte de datos según la invención;

la figura 2: un diagrama de bloques de una forma de realización a modo de ejemplo de una instalación de procesamiento de datos;

la figura 3: un diagrama de bloques de una forma de realización a modo de ejemplo de un sistema según la invención;

la figura 4a: un diagrama de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención;

la figura 4b: una matriz de confianza de una directriz de acción mínima según la segunda forma de realización preferida del dispositivo (1) de soporte de datos;

la figura 4c: una matriz de confianza de una directriz de acción media según la segunda forma de realización preferida del dispositivo (1) de soporte de datos;

la figura 4d: una matriz de confianza de una directriz de acción máxima según la segunda forma de realización preferida del dispositivo (1) de soporte de datos;

la figura 5a: un diagrama de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención;

la figura 5b: un modelo de capas de software de una forma de realización a modo de ejemplo de una instalación de procesamiento de datos;

la figura 6: un diagrama de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención;

la figura 7a: una representación esquemática del copiado de datos de un dispositivo de soporte de datos a otro dispositivo de soporte de datos;

la figura 7b: una representación esquemática del copiado de datos de un dispositivo de soporte de datos según la invención a otro dispositivo de soporte de datos;

la figura 7c: un diagrama de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención; y

la figura 7d: un pseudocódigo de una función del agente de programa.

Descripción detallada de la invención

La presente invención se describe a continuación mediante formas de realización a modo de ejemplo, en particular en forma de un dispositivo de soporte de datos según la invención, que presenta un agente de programa y datos asociados con el agente de programa.

La figura 1 muestra un diagrama de bloques de una forma de realización a modo de ejemplo de un dispositivo (1) de soporte de datos según la invención. El dispositivo (1) de soporte de datos presenta entre otros una memoria (10), que está dividida en dos zonas (11 y 12) de memoria. La memoria (10) es una memoria no volátil, por ejemplo una memoria flash.

La zona (11) de memoria está configurada como zona de memoria de sólo lectura, por ejemplo como imagen CD-ROM, y presenta un agente (11a) de programa y un agente (11b) de descifrado. La zona (12) de memoria está configurada como zona de memoria de escritura-lectura y contiene datos (12b) cifrados que están asociados con el agente (11a) de programa.

Por lo demás, el dispositivo (1) de soporte de datos presenta una unidad (13) de cifrado y/o descifrado, que puede tanto leer y descifrar datos (12b) cifrados desde la zona (12) de memoria como cifrar datos y escribirlos en la zona (12) de memoria.

A través de la interfaz (14) de datos el dispositivo (1) de soporte de datos puede unirse de manera separable con una interfaz de datos correspondiente de una instalación de procesamiento de datos (por ejemplo la interfaz (23) de datos de la instalación (2) de procesamiento de datos en la figura 2). A este respecto, a través de la interfaz (14) de datos puede accederse a la zona (11) de memoria y a la unidad (13) de cifrado y/o descifrado. La interfaz (14) de datos está configurada preferiblemente como interfaz USB y el dispositivo (1) de soporte de datos está configurado preferiblemente de manera portátil y como lápiz de memoria USB. La unión puede tener lugar de manera mecánica por ejemplo mediante inserción y es reversible.

En un primer ejemplo del dispositivo (1) de soporte de datos éste está configurado como lápiz de memoria USB, que ya en la entrega al usuario presenta el agente (11b) de descifrado y el agente (11a) de programa. Un lápiz de memoria USB de este tipo se usa debido al cifrado de los datos (12b) almacenados en el mismo en primer lugar para almacenar y transportar datos confidenciales.

Según el primer ejemplo del dispositivo (1) de soporte de datos, el agente (11a) de programa está configurado de tal manera que sólo permite acciones con los datos (12b), llamados o ejecutados por programas, que están introducidos en una denominada lista “blanca”; acciones con datos (12b) llamados o ejecutados por otros programas (distintos de los introducidos en la lista blanca) las impide el agente (11a) de programa. En el primer ejemplo del dispositivo (1) de soporte de datos, la directriz de acción está representada por tanto por la lista blanca.

Mediante este primer ejemplo se evita entre otras cosas que programas de espionaje, ejecutados en una instalación de procesamiento de datos (por ejemplo la instalación (2) de procesamiento de datos), obtengan acceso a los datos (12b), una vez que la zona (12) de memoria se ha integrado como unidad de disco en la estructura de unidad de disco de la instalación de procesamiento de datos.

En un segundo ejemplo del dispositivo (1) de soporte de datos, que establece una forma de realización preferida del dispositivo (1) de soporte de datos, éste está configurado igualmente como lápiz de memoria USB, que es parte de una solución de DLP y ya en la entrega al usuario presenta el agente de descifrado y el agente de programa. Tales soluciones de DLP evitan, por ejemplo, que datos, en particular datos confidenciales, se extraigan sin autorización de la zona controlada por las soluciones de DLP, por ejemplo mediante copiado en dispositivos de soporte de datos portátiles.

La zona controlada por una solución de DLP es en primer lugar una red de empresa fiable local con los datos confidenciales contenidos en la misma, estando instalado en cada instalación de procesamiento de datos dentro de una red de empresa de este tipo (por ejemplo instalaciones (2a-c) de procesamiento de datos locales en la figura 3) un agente de programa local, que entre otros controla que no se extraigan o se difundan sin autorización datos confidenciales de estas instalaciones de procesamiento de datos. A pesar de ello existe la necesidad de garantizar que puedan extraerse datos confidenciales de la zona controlada por la solución de DLP, por ejemplo para ofrecer la posibilidad a un trabajador de servicio externo de poder usar datos confidenciales (como listas de precios, presentaciones de clientes o similares) también fuera de la red de empresa.

Según la forma de realización preferida del dispositivo (1) de soporte de datos, el agente (11a) de programa está configurado de tal manera que sólo permite acciones con los datos (12b), que no llevan a una extracción o difusión sin autorización de los datos (12b). Es decir, el agente (11a) de programa extiende la zona controlada por la solución de DLP por la red de empresa fiable local protegiendo los datos (12b) también fuera de esta zona frente a una extracción o difusión sin autorización. Esto tiene lugar, por ejemplo, mediante una determinación en función del entorno de las acciones con datos (12b), que permite el agente (11a) de programa (directriz de acción en función del entorno).

Si la forma de realización preferida del dispositivo (1) de soporte de datos está unida con una instalación de procesamiento de datos dentro de la red de empresa (por ejemplo la instalación (2a) de procesamiento de datos propia en la figura 3), entonces el agente de programa permite todas o casi todas las acciones con datos como copiado de datos, lectura de datos, modificación de datos, impresión de datos, borrado de datos, etc. Por el contrario, fuera de la red de empresa (por ejemplo la instalación (2d o 2e) de procesamiento de datos en la figura 3) el agente (11a) de programa permite por ejemplo sólo lectura de datos. Por tanto, el agente de programa local permite sólo la transmisión de datos confidenciales dentro de la red de empresa o en dispositivos de datos, que están configurados según la forma de realización preferida del dispositivo (1) de soporte de datos.

Tanto en el primer ejemplo como en la forma de realización preferida, por ejemplo, todos los datos (12b) contenidos en la zona (12) de memoria pueden estar asociados (obligatoriamente) con el agente (11a) de programa.

En un tercer ejemplo, sólo se emula el dispositivo (1) de soporte de datos o una unión con el dispositivo (1) de soporte de datos mediante una instalación de procesamiento de datos (por ejemplo las instalaciones (2a-d) de

procesamiento de datos en la figura 3). En particular, también puede emularse una unión con el primer ejemplo o la forma de realización preferida del dispositivo (1) de soporte de datos.

Para ello se crean imágenes de memoria de las zonas (11 y 12) de memoria y se almacenan en un fichero de archivo ejecutable, iniciándose el agente de descifrado al ejecutar el fichero de archivo ejecutable.

Mediante la emulación de una unión con el dispositivo (1) de soporte de datos se imitan en relación con el software las funcionalidades del dispositivo (1) de soporte de datos, de modo que para un usuario de la instalación de procesamiento de datos durante la emulación no puede observarse ninguna diferencia con respecto a la unión con un verdadero dispositivo de soporte de datos.

La figura 2 muestra un diagrama de bloques de una forma de realización a modo de ejemplo de una instalación (2) de procesamiento de datos. La instalación (2) de procesamiento de datos puede ser tanto una instalación de procesamiento de datos propia como una instalación de procesamiento de datos central.

El procesador (20) de la instalación (2) de procesamiento de datos está configurado en particular como microprocesador, microunidad de control como microcontrolador, procesador digital de señales (DSP), circuito integrado de aplicación específica (ASIC) o disposición de puertas programables en campo (FPGA).

El procesador (20) ejecuta por ejemplo instrucciones de programa, que están almacenadas en una memoria (22) de programa, y almacena por ejemplo resultados intermedios o similares en la memoria (21) principal. Por ejemplo, la memoria (22) de programa es una memoria de sólo lectura (ROM) y la memoria (21) principal es una memoria volátil o no volátil, en particular una memoria con acceso aleatorio (RAM) y/o una memoria flash.

La memoria (22) de programa es preferiblemente un soporte de datos local unido de manera fija con la instalación (2) de procesamiento de datos. Soportes de datos unidos de manera fija con la instalación (2) de procesamiento de datos son, por ejemplo, discos duros que están incorporados en la instalación (2) de procesamiento de datos.

La memoria (22) de programa contiene el sistema operativo de la instalación (2) de procesamiento de datos, que al iniciar la instalación (2) de procesamiento de datos se carga al menos parcialmente en la memoria (21) principal y se ejecuta por el procesador (20). En particular, al iniciar la instalación (2) de procesamiento de datos se carga el núcleo del sistema operativo en la memoria (21) principal y se ejecuta por el procesador (20).

El sistema operativo de la instalación (2) de procesamiento de datos es preferiblemente un sistema operativo Windows. Un sistema operativo alternativo o adicional para la instalación (2) de procesamiento de datos es por ejemplo un sistema operativo UNIX, Linux, DOS y/o MAC.

El sistema operativo es el primero que posibilita el uso de la instalación (2) de procesamiento de datos para el procesamiento de datos. Gestiona, por ejemplo, medios operativos tales como la memoria (21) principal y memoria (22) de programa, la interfaz (23) de datos, aparatos (24) de entrada y salida, pone a disposición de otros programas entre otros mediante interfaces de programación funciones básicas y controla la ejecución de programas.

El núcleo del sistema operativo forma la capa de software más inferior de la instalación (2) de procesamiento de datos y tiene acceso directo al hardware tal como memoria (21) principal, memoria (22) de programa, interfaz (23) de datos y aparatos (24) de entrada y salida. Parte del núcleo del sistema operativo pueden ser, por ejemplo, controladores, que permiten un acceso a determinados componentes de hardware o su control. Puede no pertenecer al núcleo del sistema operativo, por ejemplo, una interfaz de usuario gráfica.

El procesador (20) controla la interfaz (23) de datos, posibilitándose el control de la interfaz (23) de datos por ejemplo mediante un controlador, que forma parte del núcleo del sistema operativo. La interfaz (23) de datos está configurada preferiblemente como interfaz USB. En particular, la interfaz (23) de datos se corresponde con la interfaz (14) de datos de tal manera, que la instalación (2) de procesamiento de datos puede unirse de manera separable con el dispositivo (1) de soporte de datos, como se explicó anteriormente.

Por lo demás, el procesador (20) controla al menos un aparato (24) de entrada/salida. El aparato (24) de entrada/salida está configurado por ejemplo como teclado, ratón, unidad de visualización, micrófono, unidad de visualización sensible al tacto, altavoz y/o cámara. El aparato (24) de entrada/salida puede registrar por ejemplo indicaciones de usuario y seguir transmitiéndolas al procesador (20) y/o recibir y emitir información para el usuario del procesador (20).

La figura 3 muestra un diagrama de bloques de una forma de realización a modo de ejemplo de un sistema (3) según la invención. El sistema (3) comprende instalaciones (2a, 2b, 2c) de procesamiento de datos propias, instalaciones (2d, 2e) de procesamiento de datos centrales, dispositivos (1a, 1d, 1e) de soporte de datos y un servidor (4). Las instalaciones (2a-2e) de procesamiento de datos corresponden a la instalación (2) de procesamiento de datos. La instalación (2a-c) de procesamiento de datos propia y el servidor (4) forman parte de la solución de DLP explicada anteriormente, que también comprende dispositivos (1a, 1d y 1e) de soporte de datos,

que están configurados según la forma de realización preferida del dispositivo (1) de soporte de datos. Los dispositivos (1a, 1d y 1e) de soporte de datos están unidos de manera separable con las instalaciones (2a, 2d y 2e) de procesamiento de datos correspondientes a través de una interfaz de datos (por ejemplo la interfaz (14) de datos en la figura 1 y la interfaz (23) de datos en la figura 2).

Las instalaciones (2a, 2b, 2c) de procesamiento de datos propias están unidas entre sí y con el servidor (4) a través de una red de empresa fiable local. En las instalaciones (2a, 2b y 2c) de procesamiento de datos propias, el agente de programa local está instalado como parte de la solución de DLP explicada anteriormente. El agente de programa local se carga al iniciar las instalaciones de procesamiento de datos propias y se ejecuta en las mismas sin interrupción.

Una vez que el agente de programa local se ejecuta en las instalaciones de procesamiento de datos propias, éste está interconectado en la comunicación interna del procesamiento de datos, es decir desvía la comunicación interna de la instalación de procesamiento de datos o partes de la comunicación interna de la instalación de procesamiento de datos a sí mismo. De este modo se desvían por ejemplo llamadas de programa, procedimiento y/o interfaz generales al agente de programa local, que se refieren por ejemplo a funciones para el copiado de datos y el movimiento de datos.

La comunicación interna de una instalación de procesamiento de datos puede referirse, por ejemplo, a la comunicación entre programas que se ejecutan en la instalación de procesamiento de datos, con otros programas y/o el sistema operativo de la instalación de procesamiento de datos; a éstos pertenecen por ejemplo las llamadas de programa, procedimiento y/o interfaz. Sin embargo, la comunicación interna de una instalación de procesamiento de datos puede referirse por ejemplo también a la comunicación entre diferentes unidades físicas de la instalación de procesamiento de datos, como memoria principal, memoria de programa, procesador y/o interfaz de datos, etc.

La desviación de llamadas de programa, procedimiento y/o interfaz se consigue por ejemplo mediante un denominado enlace API, que se describe por ejemplo en la solicitud de patente estadounidense US 2005/0108733 A1 de Microsoft Corporation. Básicamente, el “enlace API” es concebible en todos los sistemas operativos que ponen a disposición interfaces de programación. Éstos pueden ser, por ejemplo, sistemas operativos Windows, UNIX, Linux, DOS o MAC.

Una interfaz de programación de este tipo en un sistema operativo Windows es, por ejemplo, la interfaz de programación de aplicación de Windows (en inglés *Windows Application Programming Interface*, WINAPI), que pone a disposición funciones básicas para todos los programas que se ejecutan en una instalación de procesamiento de datos con un sistema operativo Windows. Funciones de la WINAPI, que se desvían al agente de programa local, son por ejemplo: CopyFile y MoveFile.

De este modo se desvían llamadas de las funciones para el copiado de datos y el movimiento de datos al agente de programa local.

La instalación (2d) de procesamiento de datos central se encuentra fuera de la red de empresa y no está unida con la red de empresa. Éste es el caso, por ejemplo, cuando un trabajador de servicio externo únicamente lleva consigo el dispositivo (1d) de soporte de datos portátil a un cliente y ejecuta una presentación (que está almacenada en la zona (12b) de memoria del dispositivo (1d) de soporte de datos) en la instalación (2d) de procesamiento de datos del cliente.

La instalación (2e) de procesamiento de datos central se encuentra igualmente fuera de la red de empresa y está unida con el servidor (4) a través de una red de área extensa (por ejemplo mediante una red privada virtual, VPN). La instalación (2e) de procesamiento de datos central es por ejemplo una instalación de procesamiento de datos portátil (por ejemplo un portátil, un PDA, un teléfono móvil, etc.) de un trabajador de servicio externo, a través de la que éste llama sus correos electrónicos desde un servidor (4).

El servidor (4) está configurado al menos parcialmente como cortafuegos y/o pasarela y une la red de empresa local con una red de área extensa como Internet. Por lo demás, en el servidor (4) pueden ejecutarse agentes adicionales, por ejemplo un agente de base de datos, agente de versión, agente de gestión y/o agente de gestión de código de descifrado, que cooperan por ejemplo con agentes en dispositivos (1a, 1d, 1e) de soporte de datos y/o instalaciones (2a-2e) de procesamiento de datos, por ejemplo para actualizar el agente de programa.

La figura 4a muestra un diagrama (400) de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención, que se ejecutan cada vez que se une un dispositivo (1) de soporte de datos según la invención con una instalación (2) de procesamiento de datos. Esto se refiere en particular también a la unión del primer ejemplo o de la forma de realización preferida del dispositivo (1) de soporte de datos con una instalación (2) de procesamiento de datos (por ejemplo la instalación (2a, 2d, 2e) de procesamiento de datos en la figura 3).

En la etapa (401) se produce la unión del dispositivo (1) de soporte de datos con la instalación (2) de procesamiento de datos, insertando el usuario por ejemplo la interfaz (14) de datos, que está configurada como interfaz USB, en la interfaz (23) de datos correspondiente a la misma.

En la etapa (402) el sistema operativo de la instalación (2) de procesamiento de datos reconoce que a través de la interfaz (23) de datos se estableció una unión con el dispositivo (1) de soporte de datos, e integra la zona (11) de memoria como nueva unidad de disco en la estructura de unidad de disco de la instalación (2) de procesamiento de datos.

Si el dispositivo (1) de soporte de datos está configurado por ejemplo como lápiz de memoria USB (por ejemplo según el primer ejemplo y/o la forma de realización preferida del dispositivo (1) de soporte de datos), entonces la integración de la zona (11) de memoria puede producirse por ejemplo mediante un controlador, que se carga en relación con la norma junto con el núcleo del sistema operativo o está contenido en el mismo y que pone a disposición en relación con la norma los datos del sistema operativo contenidos en lápices de memoria USB. Una instalación de un controlador no sería necesaria en este caso, la integración de la zona (11) de memoria tendría lugar sin necesidad de instalación. En consecuencia, también todos los programas ejecutables, que están almacenados en la zona (11) de memoria, pueden ejecutarse sin necesidad de instalación.

Por el contrario, el sistema operativo aún no puede disponer de los datos (12b) contenidos en la zona (12) de memoria, puesto que éstos están cifrados y la unidad (13) de cifrado y/o descifrado evita todo acceso a los mismos.

En la etapa (403), el usuario inicia el agente (11b) de descifrado, que está configurado como programa informático ejecutable y se encuentra en la unidad de disco recién integrada. Alternativamente es concebible, por ejemplo, que el agente (11b) de descifrado se inicie automáticamente por el sistema operativo, una vez que la zona (11) de memoria se integra como nueva unidad de disco en la estructura de unidad de disco de la instalación (2) de procesamiento de datos (por ejemplo a través de la funcionalidad Autorun del sistema operativo Windows).

Al iniciar el agente (11b) de descifrado se carga su código de programa a través de la interfaz (23) de datos (por ejemplo directamente) en la memoria (21) principal y se ejecuta por el procesador (20). El agente (11b) de descifrado se ejecuta tras el inicio por ejemplo como proceso en la instalación (2) de procesamiento de datos.

Una vez que el agente (11b) de descifrado se ejecuta en la instalación (2) de procesamiento de datos, inicia en la etapa (404) el agente (11a) de programa. Es decir, el agente (11b) de descifrado provoca que el agente (11a) de programa (por ejemplo a través de la interfaz (23) de datos) se cargue en la memoria (21) principal y se ejecute en el procesador (20). El agente (11a) de programa se ejecuta tras el inicio por ejemplo como parte del proceso del agente (11b) de descifrado o como proceso propio en la instalación (2) de procesamiento de datos.

Una vez que el agente (11a) de programa se ejecuta en la instalación (2) de procesamiento de datos, se introduce en la etapa (405) en la comunicación interna de la instalación de procesamiento de datos y desvía llamadas de interfaces de programación, que se ejecutan como procesos en la instalación (2) de procesamiento de datos y ponen a disposición de otros programas funciones, que ejecutan acciones con datos, a sí mismo. Es decir, tras introducirse, el agente de programa está interconectado en la comunicación interna de la instalación de procesamiento de datos.

Normalmente, las interfaces de programación están configuradas como bibliotecas dinámicas y se cargan por ejemplo como parte del sistema operativo al iniciar la instalación (2) de procesamiento de datos en la memoria (21) principal y se ejecutan por el procesador (20). Las interfaces de programación se ejecutan en la instalación (2) de procesamiento de datos como procesos.

Tales interfaces de programación en un sistema operativo Windows son, por ejemplo, la interfaz de programación de aplicación de Windows (en inglés *Windows Application Programming Interface*, WINAPI) y la interfaz de programación de zócalo de Windows (en inglés *Windows Socket Application Programming Interface*, Winsock-API).

La WINAPI pone a disposición funciones básicas para todos los programas que se ejecutan en una instalación de procesamiento de datos con un sistema operativo Windows. Funciones de la WINAPI, cuyas llamadas se desvían al agente (11a) de programa, son por ejemplo: CopyFile, MoveFile, StartDoc, StartDocPrinter, CreateFile y GetClipBoardData.

La Winsock-API pone a disposición funciones para el acceso a componentes de red para todos los programas que se ejecutan en una instalación de procesamiento de datos con un sistema operativo Windows. Funciones de la Winsock-API, cuyas llamadas se desvían al agente (11a) de programa, son por ejemplo: send y sendto.

De este modo se desvían llamadas de las funciones, que ejecutan acciones con datos, al agente (11a) de programa.

Según la forma de realización preferida del dispositivo (1) de soporte de datos, tras haberse configurado las desviaciones descritas, se determina la directriz de acción en función del entorno. La directriz de acción fija qué acciones con datos (12b), que están asociadas con el agente (11a) de programa, se permiten o se impiden.

Para la determinación se compara el entorno de la instalación (2) de procesamiento de datos (en la que se ejecuta el agente (11a) de programa) con perfiles de entorno establecidos y se selecciona la directriz de acción, que está asignada al perfil de entorno con las coincidencias exactas o las mayores. Si las coincidencias con los perfiles de entorno establecidos son demasiado pequeñas, entonces puede seleccionarse por ejemplo una directriz de acción mínima.

El entorno de la instalación (2) de procesamiento de datos se refiere a al menos su ubicación o conexiones de red.

La instalación (2a) de procesamiento de datos propia (en la figura 3) se encuentra por ejemplo dentro de la red de empresa fiable local, de modo que en este caso se selecciona una directriz de acción máxima. Según la directriz de acción máxima se permiten todas o casi todas las acciones con datos (12b) del dispositivo (1a) de soporte de datos.

La instalación (2d) de procesamiento de datos central se encuentra fuera de la red de empresa fiable local, de modo que en este caso se selecciona una directriz de acción mínima. Según la directriz de acción mínima sólo se permiten acciones con datos (12b) del dispositivo (1d) de soporte de datos como lectura de datos.

La instalación (2e) de procesamiento de datos central se encuentra fuera de la red de empresa fiable local, pero está unida a través de una red de área extensa con el servidor (4) (por ejemplo mediante una red privada virtual, VPN), de modo que en este caso se selecciona una directriz de acción media. Según la directriz de acción media se permiten por ejemplo sólo acciones con datos (12b) del dispositivo (1e) de soporte de datos como lectura de datos y la transferencia de datos (12b) a o a través del servidor (4) (por ejemplo a través de la VPN).

Las directrices de acción en función del entorno están configuradas por ejemplo como matriz (por ejemplo matriz (4b, 4c y 4d) de confianza en las figuras 4b-d), de la que puede deducirse qué acción con cuáles de los datos (12b) se permitirá y cuál no.

A diferencia de esto, en el primer ejemplo del dispositivo (1) de soporte de datos no es necesaria una determinación de la directriz de acción, puesto que la lista blanca es independiente del entorno de la instalación de procesamiento de datos en la que se ejecuta el agente (11a) de programa.

En la etapa (406), el usuario introduce su código de descifrado, después de que se lo pida el agente (11b) de descifrado.

En la etapa (407), el agente (11b) de descifrado sigue transmitiendo el código de descifrado introducido por el usuario a la unidad (13) de cifrado y/o descifrado, que comprueba el código de descifrado y, si es correcto, libera la clave para el descifrado de los datos cifrados en la zona (12) de memoria.

Mediante la liberación de la clave se ponen los datos cifrados en la zona (12) de memoria a disposición del sistema operativo, que a continuación integra la zona (12) de memoria como nueva unidad de disco en la estructura de unidad de disco de la instalación (2) de procesamiento de datos.

Un acceso a los datos (12b) cifrados se produce ahora de manera transparente, es decir se descifran mediante la unidad (13) de cifrado y/o descifrado automáticamente a demanda (sobre la marcha). Por tanto, al acceder a los datos (12b) en la zona (12) de memoria no puede observarse ninguna diferencia con el acceso a datos no cifrados en otras unidades de disco (por ejemplo la zona (11) de memoria).

Según el tercer ejemplo, al ejecutar el fichero de archivo ejecutable se inicia el agente (11b) de descifrado, esto corresponde a la etapa (403) en el diagrama (400) de flujo. Las etapas (401 y 402) anteriores no son necesarias según el tercer ejemplo, puesto que el fichero de archivo ejecutable ya se encuentra en una memoria (por ejemplo la memoria (22) de programa) de la instalación (2) de procesamiento de datos. Por ejemplo, el fichero de archivo ejecutable se ha transmitido por correo electrónico a la instalación (2) de procesamiento de datos. Tras el inicio del agente (11b) de descifrado se ejecutan las etapas (404-406) como en el diagrama (400) de flujo.

Sin embargo, la etapa (407) se modifica ligeramente según el tercer ejemplo, y concretamente el agente (11b) de descifrado comprueba el código de descifrado introducido y, si es correcto, hace que la imagen de memoria de la zona (12) de memoria (es decir los datos (12b) cifrados) esté disponible para el sistema operativo de tal manera que la imagen de memoria se integre como nueva unidad de disco virtual en la estructura de unidad de disco de la instalación (2) de procesamiento de datos. El acceso a los datos (12b) se produce ahora igualmente de manera transparente, y el usuario no puede observar ninguna diferencia con el acceso a datos en otras unidades de disco (reales).

Las figuras 4b-d muestran a modo de ejemplo las matrices (4b-d) de confianza de una directriz de acción mínima, media y máxima según la forma de realización preferida del dispositivo (1) de soporte de datos.

Las matrices de confianza indican si debe permitirse o no una determinada acción con determinados datos de los datos (12b) cifrados.

En las matrices (4b-d) de confianza, a cada archivo (archivos (12b-1 - 12b-15)) está asignada como elemento más pequeño de los datos (12b) cifrados una línea y a cada acción una columna, estableciendo el enlace en la celda, que pertenece a una acción y a un archivo, si esta acción con este archivo incumple o no la respectiva directriz de acción, es decir si debe impedirse o permitirse.

El enlace "+" representa a este respecto una acción con datos, que no incumple la directriz de acción; y el enlace "-" representa a este respecto una acción con datos, que incumple la directriz de acción.

Según la matriz (4b) de confianza de la directriz de acción máxima se permiten por ejemplo todas las acciones con los archivos (12b-1 - 12b-15). A diferencia de esto, la matriz (4d) de confianza de la directriz de acción mínima sólo permite la acción de lectura de datos para todos los archivos (12b-1 - 12b-15) y evita todas las demás acciones.

Según la matriz (4c) de confianza de la directriz de acción media se permitirá por ejemplo la acción de lectura de datos para todos los archivos (12b-1 - 12b-15). La acción de borrado de datos se impedirá en este caso para todos los archivos (12b-1 - 12b-15). Por lo demás, pueden copiarse los archivos (12b-1 - 12b-3), sin embargo los archivos (12b-4 - 12b-15) no. Una clasificación de este tipo es útil, por ejemplo, para posibilitar a un trabajador de servicio externo la transmisión de archivos (12b-1 - 12b-3) poco o no confidenciales. Para autorizar el intercambio de datos con colegas dentro de la red de empresa fiable local, debe permitirse además la transferencia de todos los archivos (12b-1 - 12b-15) a través de una VPN con el servidor (4).

Básicamente, las directrices de acción (por ejemplo matrices (4b-4d) de confianza, lista blanca, etc.) pueden ser, por ejemplo, parte del agente de programa y están predefinidas, por ejemplo, por un administrador y se adaptan también por éste. La adaptación puede producirse por ejemplo a través de una operación de actualización mediante un programa correspondiente (por ejemplo un agente de actualización) en el servidor (4) y/o el dispositivo (1) de soporte de datos.

La figura 5a muestra un diagrama (500) de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención, que se ejecutan una vez que debe ejecutarse una acción con datos en una instalación de procesamiento de datos, en la que se ejecuta el agente de programa según la invención. La instalación de procesamiento de datos puede ser, por ejemplo, una instalación de procesamiento de datos propia (por ejemplo la instalación (2a) de procesamiento de datos propia en la figura 3) o una instalación de procesamiento de datos central (por ejemplo la instalación (2e y 2d) de procesamiento de datos central en la figura 3), que está unida de manera separable con el dispositivo (1) de soporte de datos.

Las etapas del diagrama (500) de flujo se ejecutan por el agente (11a) de programa, en particular por el agente (11a) de programa, que está contenido en un dispositivo de soporte de datos según el primer ejemplo del dispositivo (1) de soporte de datos, y por el agente (11a) de programa, que está contenido en un dispositivo de soporte de datos según la forma de realización preferida del dispositivo (1) de soporte de datos.

A continuación se parte de la base de que el agente (11a) de programa se ejecuta en la instalación (2) de procesamiento de datos, que se unió con el dispositivo (1) de soporte de datos y en la que ya se han ejecutado las etapas según el diagrama (400) de flujo o que emula una unión con un dispositivo (1) de soporte de datos según el tercer ejemplo.

En la etapa (501) se intercepta una llamada de una función, que realiza una acción con datos. Interceptar significa en este caso, que la llamada de la función se desvió al agente (11a) de programa y que la acción con los datos no se realiza por la función.

La función puede ser una de las funciones descritas anteriormente de interfaces de programación del sistema operativo Windows, sin embargo a este respecto también puede tratarse de otras funciones.

En particular se interceptan llamadas de funciones, que ejecutan las siguientes acciones con datos (12b):

- Copiado y/o movimiento de datos: para evitar un copiado no deseado de datos (12b) asociados con el agente (11a) de programa y/o de datos en el dispositivo de soporte de datos, se interceptan todas las llamadas correspondientes de funciones de copiado de datos y movimiento de datos. Éstas son por ejemplo llamadas de las funciones CopyFile y MoveFile de la WINAPI del sistema operativo Windows. A este respecto se interceptan todas las llamadas de funciones correspondientes, que parten de programas que se ejecutan en la instalación (2) de procesamiento de datos.

- Impresión de datos: para evitar una impresión no deseada de datos (12b) asociados con el agente (11a) de programa, se interceptan todas las llamadas correspondientes de funciones de impresión de datos. Éstas son por ejemplo llamadas de las funciones StartDoc y StartDocPrinter de la WINAPI del sistema operativo

Windows. A este respecto se interceptan todas las llamadas de funciones correspondientes, que parten de programas que se ejecutan en la instalación (2) de procesamiento de datos y por ejemplo están introducidos en una lista de usuario. En la lista de usuario están introducidos programas y/o procesos que se ejecutan en la instalación (2) de procesamiento de datos y realizan acciones con los datos (12b) asociados con el agente (11a) de programa.

- Almacenamiento de datos, almacenamiento de datos como y/o apertura de datos: para evitar una apertura, marcación, edición y/o almacenamiento no deseados de datos (12b) asociados con el agente (11a) de programa, se interceptan todas las llamadas correspondientes de funciones de almacenamiento de datos y/o almacenamiento de datos como. Éstas son por ejemplo llamadas de la función CreateFile de la WINAPI del sistema operativo Windows. A este respecto se interceptan todas las llamadas de funciones correspondientes, que parten de programas que se ejecutan en la instalación (2) de procesamiento de datos.

- Transferencia de datos a través de redes: para evitar una transferencia no deseada de datos (12b) asociados con el agente (11a) de programa a través de una red, por ejemplo Internet y/o cualquier otra red, se interceptan todas las llamadas correspondientes de funciones de transferencia de datos a través de redes. Éstas son por ejemplo llamadas de las funciones send y sendto de la Winsock-API del sistema operativo Windows. A este respecto se interceptan todas las llamadas de funciones correspondientes, que parten de programas que se ejecutan en la instalación (2) de procesamiento de datos.

- Intercambio de datos a través de comunicación entre procesos: para evitar un intercambio no deseado de datos (12b) asociados con el agente (11a) de programa entre programas, que se han introducido en la lista de usuario, y otros programas, que se ejecutan en la instalación (2) de procesamiento de datos, se interceptan todas las llamadas correspondientes de funciones de intercambio de datos a través de comunicación entre procesos (en inglés *Inter-Process Communication*, IPC). Éstas son por ejemplo llamadas de la función GetClipboardData de la WINAPI del sistema operativo Windows. Por lo demás, se interceptan por ejemplo también llamadas de funciones, que introducen programas en la lista de usuario. A este respecto se interceptan todas las llamadas de funciones correspondientes, que parten de programas que se ejecutan en la instalación (2) de procesamiento de datos.

En la llamada se transfieren a la función, que realiza una acción con datos (12b), los datos como argumento. Un argumento de este tipo puede ser, por ejemplo, un parámetro de referencia como un indicador en un mapa de memoria que se encuentra en la memoria (21) principal y/o un parámetro de valor.

En la etapa (502), el agente (11a) de programa comprueba la llamada interceptada de una función, que realiza una acción con datos, en cuanto a si la acción con los datos incumple una directriz de acción.

Para ello se verifica en primer lugar si debe realizarse la acción con datos, que están asociados con el agente (11a) de programa.

Por ejemplo según el primer ejemplo y la forma de realización preferida del dispositivo (1) de soporte de datos, todos los datos (12b) contenidos en la zona (12) de memoria están asociados con el agente (11a) de programa. Si la llamada interceptada contiene como argumento por ejemplo una ruta, que remite a datos (12b) en la zona (12) de memoria del dispositivo (1) de soporte de datos, entonces los datos contenidos como argumento tanto según el primer ejemplo como según la forma de realización preferida del dispositivo (1) de soporte de datos están asociados con el agente (11a) de programa. Esto puede observarse por ejemplo mediante la letra de unidad de disco contenida en la ruta, cuando la zona (12) de memoria (es decir los datos asociados con el agente (11a) de programa) se integró como nueva unidad de disco (independiente) en la estructura de unidad de disco de la instalación (2) de procesamiento de datos.

Además, la posición de los mapas de memoria en la memoria (21) principal puede protocolizarse por datos asociados con el agente (11a) de programa, que pertenecen a procesos de programas en la lista de usuario, y compararse con un argumento de la llamada interceptada. Alternativamente, por ejemplo es concebible que las llamadas interceptadas, que parten de programas y/o llaman programas, que se han introducido en la lista de usuario, siempre se refieran a datos que están asociados con el agente (11a) de programa. Esto es ventajoso, por ejemplo, en el caso del primer ejemplo del dispositivo (1) de soporte de datos, para aislar los programas introducidos en la lista de usuario de programas de espionaje (que posiblemente se ejecutan en la instalación de procesamiento de datos).

Acciones con datos, que no están asociados con el agente (11a) de programa, básicamente no incumplen la directriz de acción y tampoco se comprueban en este sentido.

Según el primer ejemplo del dispositivo (1) de soporte de datos, todas las acciones con datos (12b), que se llaman o realizan por programas que no están introducidos en la lista blanca, incumplen la directriz de acción. Sin embargo es concebible, que se pida al usuario que indique si una acción con datos (12b), que se llama o realiza por un programa

que no está introducido en la lista blanca, debe permitirse a pesar de ello. Una indicación de usuario de este tipo puede almacenarse por ejemplo en la lista blanca.

Por tanto, la verificación según este primer ejemplo puede realizarse mediante una comparación del nombre del programa que llama o llamado de la llamada interceptada con los nombres de programa en la lista blanca.

Según la forma de realización preferida del dispositivo (1) de soporte de datos, el agente de programa verifica si la acción con los datos incumple la directriz de acción en función del entorno determinada previamente. Esto tiene lugar por ejemplo mediante una comparación de la acción y de los datos con una matriz tal como la matriz (4c) de confianza en la figura 4b.

Si la verificación en la etapa (502) da como resultado que la llamada interceptada de la función, que realiza una acción con los datos, no incumple ninguna directriz de acción, entonces se realiza la etapa (503).

En la etapa (503) se sigue transmitiendo la llamada desviada al agente (11a) de programa a la verdadera función, de modo que ésta realiza la acción con los datos. Al mismo tiempo se introduce en la lista de usuario el programa del que partió la llamada de la función, en caso de utilizar una.

Si la verificación en la etapa (502) da como resultado que la llamada interceptada de la función, que realiza una acción con los datos, incumple una directriz de acción, entonces se realiza la etapa (504).

En la etapa (504) se le indica al usuario, que la acción con los datos incumple una directriz de acción, y en la etapa (505) se interrumpe o se finaliza la llamada desviada al agente (11a) de programa. Una llamada puede finalizarse, por ejemplo, al no seguir transmitiéndola.

Por lo demás, a las etapas (503 y 505) puede seguir además una protocolización de la verificación mediante un agente de protocolización, en caso de que éste se ejecute en la instalación de procesamiento de datos.

La figura 5b muestra un modelo de capas de software de una forma de realización a modo de ejemplo de la instalación (2) de procesamiento de datos, en la que se ejecuta el agente (11a) de programa. La instalación de procesamiento de datos puede ser por ejemplo una instalación de procesamiento de datos propia (por ejemplo la instalación (2a) de procesamiento de datos propia en la figura 3) o una instalación de procesamiento de datos central (por ejemplo la instalación (2e y 2d) de procesamiento de datos central en la figura 3), que está unida de manera separable con el dispositivo (1) de soporte de datos.

El modelo de capas de software es válido para todas las instalaciones de procesamiento de datos en las que se ejecuta el agente (11a) de programa, en particular instalaciones de procesamiento de datos en las que se ejecuta el agente (11a) de programa que está contenido en un dispositivo de soporte de datos según el primer ejemplo del dispositivo (1) de soporte de datos, y/o el agente (11a) de programa que está contenido en un dispositivo de soporte de datos según la forma de realización preferida del dispositivo (1) de soporte de datos.

El modelo de capas de software muestra a modo de ejemplo tres capas de software, concretamente una capa (50) intermedia con el agente (11a) de programa, una capa (51) inferior con el sistema operativo y una capa (53) superior con aplicaciones.

La capa (51) inferior pone a disposición de las aplicaciones de la capa (53) superior funciones generales, por ejemplo a través de interfaces de programación. Una función (52) de este tipo se representa en la figura 5b a modo de ejemplo.

Aplicaciones de la capa (53) superior se muestran en la figura 5b como programas (54, 55 y 56), que se ejecutan en la instalación de procesamiento de datos.

En la figura 5b puede reconocerse claramente que el agente (11a) de programa intercepta tanto una llamada de la función (52) mediante el programa (56) como una llamada de una función del programa (55) mediante el programa (54) o un intercambio de datos entre el programa (54 y 55) y no las permite hasta una verificación satisfactoria.

La figura 6 muestra un diagrama (600) de flujo con etapas de procedimiento de una forma de realización a modo de ejemplo de la invención, que se realizan una vez que deban copiarse datos de la instalación (2a) de procesamiento de datos propia en el dispositivo (1a) de soporte de datos.

Las etapas del diagrama (600) de flujo se realizan por el agente de programa local según la solución de DLP explicada anteriormente (que comprende dispositivos de soporte de datos según la forma de realización preferida del dispositivo (1) de soporte de datos), que se instaló en la instalación de procesamiento de datos propia y se ejecuta en la misma.

En la etapa (601) se intercepta en primer lugar una llamada de una función de una interfaz de programación para copiar y/o mover datos. Interceptar significa en este caso que la llamada de la función se desvía al agente de programa local y que el copiado y/o movimiento de datos no se realiza por la función.

Si los datos sólo deben copiarse y/o moverse a un soporte de datos local, como la memoria (22) de programa, unido de manera fija con la instalación de procesamiento de datos, entonces la llamada desviada al agente de programa local se sigue transmitiendo directamente a la función llamada, que copia y/o mueve los datos. Para ello se comprueba por ejemplo la letra de unidad de disco de la ruta de destino.

Sin embargo, si los datos deben copiarse en el dispositivo (1a) de soporte de datos portátil, que está unido de manera separable con la instalación (2a) de procesamiento de datos, entonces en la etapa (602) se comprueba si los datos, cuando se copiaron en el dispositivo (1a) de soporte de datos, se asocian con el agente (11a) de programa. Para ello se comprueba si el dispositivo (1a) de soporte de datos corresponde a la forma de realización preferida del dispositivo (1) de soporte de datos.

Si el dispositivo (1a) de soporte de datos corresponde a la forma de realización preferida del dispositivo (1) de soporte de datos, entonces en la etapa (603) se sigue transmitiendo la llamada desviada al agente de programa local a la verdadera función, de modo que ésta copia y/o mueve los datos al dispositivo (1a) de soporte de datos.

Sin embargo, si el dispositivo (1a) de soporte de datos no corresponde a la forma de realización preferida del dispositivo (1) de soporte de datos, y los datos tampoco se asocian de otra manera con el agente (11a) de programa cuando se copiaron en el dispositivo (1a) de soporte de datos, entonces se realiza la etapa (604).

En la etapa (604) se indica al usuario que el copiado y/o movimiento de los datos no es posible, y a continuación en la etapa (605) se interrumpe o finaliza la llamada desviada al agente de programa local.

La figura 7a muestra una representación esquemática del copiado de datos de un dispositivo de soporte de datos portátil conocido en el estado de la técnica, que está unido con la instalación (2) de procesamiento de datos, en la memoria (22) de programa de la instalación (2) de procesamiento de datos.

La memoria (70) del dispositivo de soporte de datos portátil se ha integrado mediante el sistema operativo de la instalación (2) de procesamiento de datos como unidad de disco con la ruta G:\. La memoria (22) de programa tiene la ruta C:\.

Mediante la orden (71) (“Copy G:\Myfile.txt C:\CopyOfMyFile.txt”) se copiará el archivo MyFile.txt de la memoria (70) en la memoria (22) de programa como CopyOfMyFile.txt. La orden puede introducirse por ejemplo por un usuario y/o un programa.

La orden (71) llama una función RealCopyFile (72a), que se pone a disposición por una interfaz de programación a la capa (51) inferior y realiza el copiado de datos de una ruta de origen a una ruta de destino. A este respecto, la función RealCopyFile (72a) llama funciones adicionales, que se ponen a disposición por el núcleo (73) del sistema operativo, de modo que se copian los datos según la orden (71).

La figura 7b muestra una representación esquemática del copiado de datos (12b) del primer ejemplo o de la forma de realización preferida del dispositivo (1) de soporte de datos, que está unida con la instalación (2) de procesamiento de datos, en la memoria (22) de programa de la instalación (2) de procesamiento de datos; y en la figura 7c se muestra el diagrama (700) de flujo correspondiente a la figura 7b.

A continuación se parte de que el agente (11a) de programa se ejecuta en la instalación (2) de procesamiento de datos, que se unió con el primer ejemplo o la forma de realización preferida del dispositivo (1) de soporte de datos y en la que ya se han realizado las etapas según el diagrama (400) de flujo.

La zona (12) de memoria del dispositivo (1) de soporte de datos se ha integrado mediante el sistema operativo de la instalación (2) de procesamiento de datos como unidad de disco con la ruta G:\. La memoria (22) de programa tiene la ruta C:\.

En la etapa (701) mediante la orden (71) (“Copy G:\Myfile.txt C:\CopyOfMyFile.txt”) se copiará el archivo MyFile.txt de la zona (12) de memoria en la memoria (22) de programa como CopyOfMyFile.txt, como en la figura 7a.

La orden (71) llama la función RealCopyFile (72a), cuya llamada sin embargo en la etapa (702) se desvía a la función MyCopyFile (72b) del agente (11a) de programa.

En la etapa (703) la función MyCopyFile (72b) verifica si el copiado del archivo MyFile.txt de la zona (12) de memoria en la memoria (22) de programa como CopyOfMyFile.txt incumple la directriz de acción. Mediante la ruta de origen puede reconocerse que el archivo MyFile.txt está asociado con el agente (11a) de programa, es decir, que pertenece a los datos (12b) cifrados.

En caso de que el copiado no incumpla la directriz de acción, en la etapa (704) se sigue transmitiendo la llamada desviada, como se muestra en la figura 7b, a la función RealCopyFile (72a), que copia los datos según la orden (71) o provoca el copiado.

Por ejemplo, según el primer ejemplo no existe ningún incumplimiento de la directriz de acción cuando tanto la orden (71) de un programa como la interfaz de programación, que pone a disposición la función RealCopyFile (72a), están introducidas en la lista blanca.

En el caso de la forma de realización preferida no existe ningún incumplimiento de la directriz de acción, cuando debe permitirse el copiado del archivo MyFile.txt según la matriz de confianza de la directriz de acción determinada. En caso de que el archivo MyFile.txt corresponda por ejemplo al archivo (12b-1) en la matriz (4c) de confianza y se haya determinado la directriz de acción media, entonces el copiado del archivo My-File.txt no incumpliría la directriz de acción.

Si, por el contrario, existe un incumplimiento de la directriz de acción, en la etapa (705) se muestra que no es posible el copiado de los datos y la llamada desviada no se sigue tratando.

Según el primer ejemplo existe un incumplimiento de la directriz de acción por ejemplo cuando al menos la orden (71) de un programa o la interfaz de programación, que pone a disposición la función RealCopyFile (72a), no está introducida en la lista blanca.

En el caso de la forma de realización preferida existe un incumplimiento de la directriz de acción, cuando debe impedirse el copiado del archivo MyFile.txt según la matriz de confianza de la directriz de acción determinada. En caso de que el archivo MyFile.txt corresponda por ejemplo al archivo (12b-8) en la matriz (4c) de confianza y se haya determinado la directriz de acción media, entonces el copiado del archivo MyFile.txt incumpliría la directriz de acción.

La figura 7d muestra un pseudocódigo de la función MyCopyFile (72b) del agente (11a) de programa.

La secuencia de las etapas de procedimiento individuales en los diagramas de flujo individuales no es obligatoria, son concebibles secuencias alternativas de las etapas de procedimiento. Las etapas de procedimiento pueden implementarse de diferente manera, así es concebible una implementación en software (mediante instrucciones de programa), hardware o una combinación de ambas para la implementación de las etapas de procedimiento.

REIVINDICACIONES

1. Procedimiento, que comprende:

- determinar una directriz de acción mediante un agente (11a) de programa y/o mediante un agente de perfil llamado por el agente de programa, cuando un aparato electrónico se une de manera separable con una instalación de procesamiento de datos, en el que el agente (11a) de programa está almacenado en el aparato electrónico y se ejecuta directamente desde el aparato (1, 1a, 1d, 1e) electrónico en la instalación (2, 2a-e) de procesamiento de datos,

  en el que determinar la directriz (4) de acción comprende seleccionar una directriz (4) de acción a partir de varias directrices (4b, 4c, 4d) de acción establecidas y se basa al menos parcialmente en un entorno, en el que se ejecuta el agente de programa, y

  en el que el entorno comprende la ubicación de la instalación de procesamiento de datos, su conexión de red y/o las aplicaciones y/o procesos instalados y/o que se ejecutan en la misma,

- verificar mediante el agente (11a) de programa, si acciones (71) con datos (12b) incumplen la directriz de acción determinada (502, 703), en el que las acciones (71) con datos (12b) comprenden borrado de datos, lectura de datos, modificación de datos, copiado de datos, impresión de datos, almacenamiento de datos, transferencia de datos a través de redes, transferencia de datos a través de una red privada virtual o intercambio de datos a través de comunicación entre procesos, en el que el agente (11a) de programa está asociado con los datos (12b) y los datos (12b) asociados con el agente (11a) de programa están almacenados junto con el agente (11a) de programa en el aparato (1, 1a, 1d, 1e) electrónico, en el que los datos (12b) en el aparato (1, 1a, 1d, 1e) electrónico a través del lugar de almacenamiento común están asociados con el agente (11a) de programa,

- permitir (503, 704) las acciones mediante el agente (11a) de programa, en caso de que no incumplan la directriz (4) de acción determinada, e impedir (504-505, 705) las acciones mediante el agente (11a) de programa, en caso de que incumplan la directriz (4) de acción determinada.

2. Procedimiento según la reivindicación 1, en el que determinar la directriz (4) de acción comprende además comparar el entorno de la instalación (2) de procesamiento de datos con varios perfiles de entorno establecidos, en el que a cada perfil de entorno está asignada una de las directrices de acción, y en el que se selecciona la directriz de acción, que está asignada al perfil de entorno con la mayor coincidencia con el entorno de la instalación (2) de procesamiento de datos.

3. Procedimiento según una de las reivindicaciones 1-2, en el que las acciones (71) con los datos (12b) están configuradas como llamada de programa, procedimiento y/o interfaz, a la que como argumento se transfieren datos (12b) asociados con el agente (11a) de programa, y/o intentos de acceso a datos (12b) asociados con el agente (11a) de programa.

4. Procedimiento según una de las reivindicaciones 1-3, en el que los datos (12b) asociados con el agente (11a) de programa están almacenados en el aparato (1, 1a, 1d, 1e) electrónico de manera cifrada, en particular de tal manera que los datos (12b) asociados con el agente (11a) de programa sólo se descifran en caso de que la acción (71) no incumpla ninguna directriz (4) de acción.

5. Procedimiento según una de las reivindicaciones 1-4, que comprende además:

- introducir programas (54-56), que se ejecutan en la instalación (2, 2a-e) de procesamiento de datos y que realizan al menos una acción con los datos (71), en una lista de usuario y

- finalizar todos los programas (54-56) que se ejecutan en la instalación (2, 2a-e) de procesamiento de datos, que se han introducido en la lista de usuario, antes de finalizar el agente (11a) de programa.

6. Programa (11a) informático, que comprende instrucciones de programa que provocan que un procesador (20) realice el procedimiento según una de las reivindicaciones 1-5, cuando el programa (11a) informático se ejecuta en el procesador (20).

7. Dispositivo (1, 1a, 1d, 1e) de soporte de datos, en el que está almacenado un programa (11a) informático según la reivindicación 6.

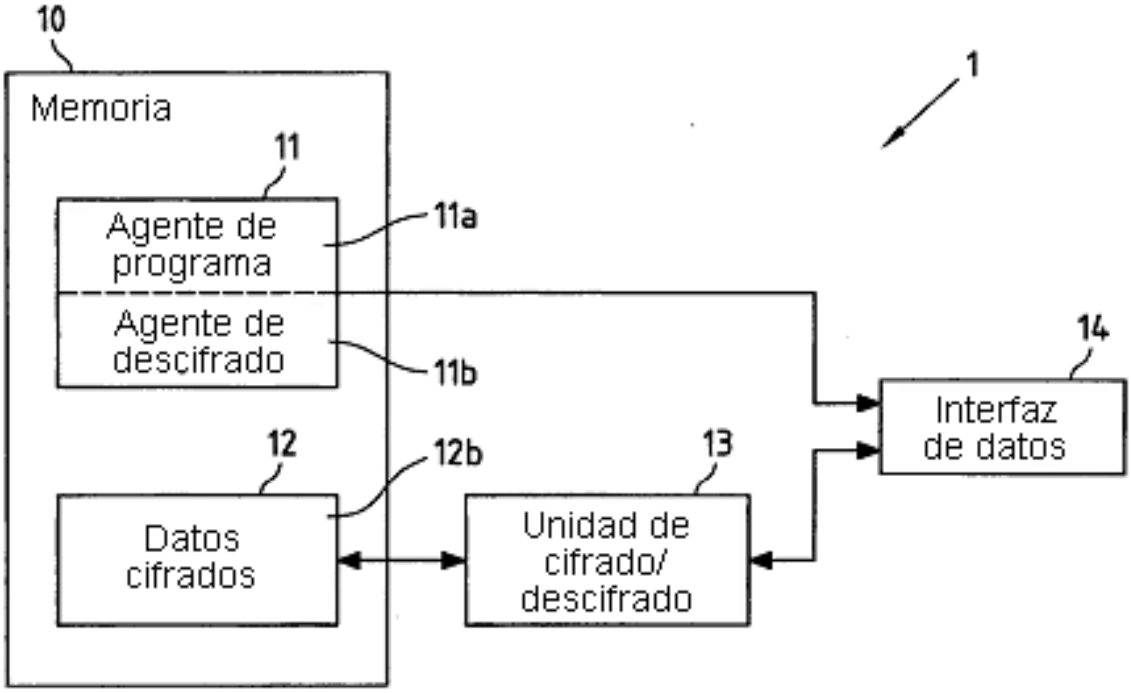
10
Memoria
11
Agente de programa
11a
Agente de descifrado
11b
1
14
Interfaz de datos
12
12b
13
Datos cifrados
Unidad de cifrado/ descifrado

Fig.1

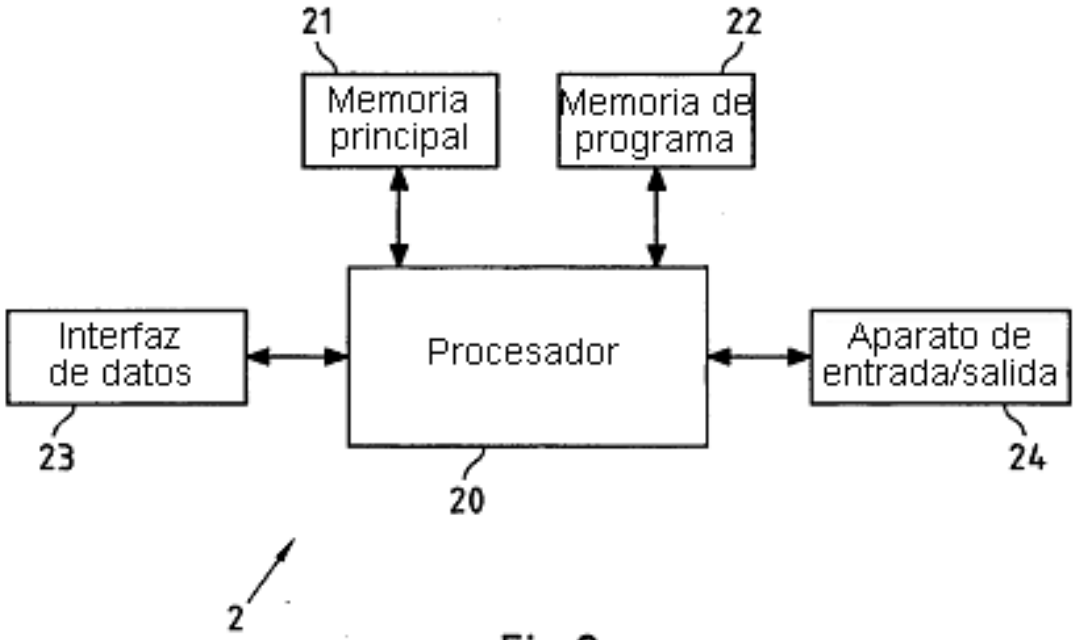
21
Memoria principal
22
Memoria de programa
Interfaz de datos
Procesador
Aparato de entrada/salida
23
20
24
2

Fig.2

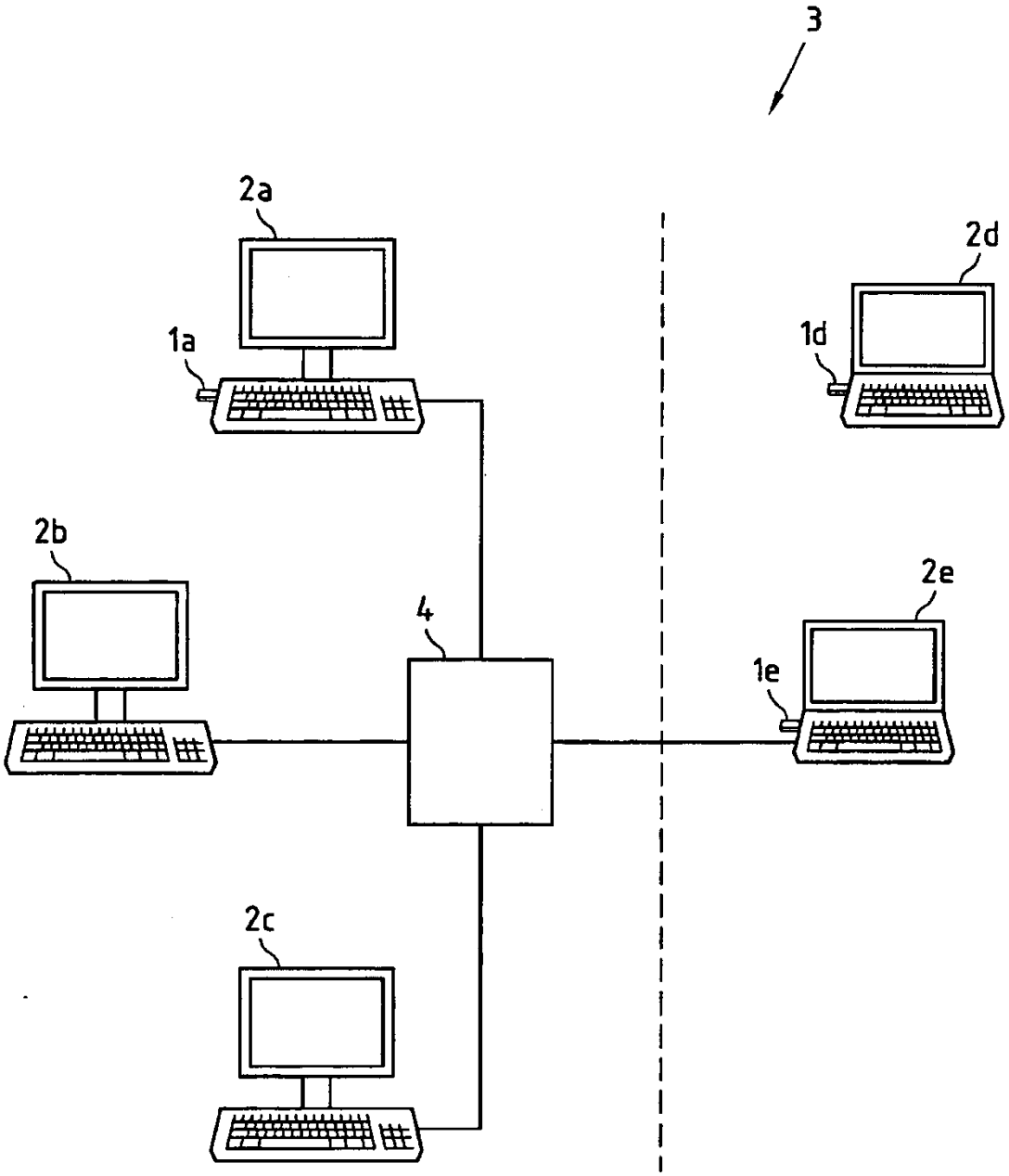

Fig.3

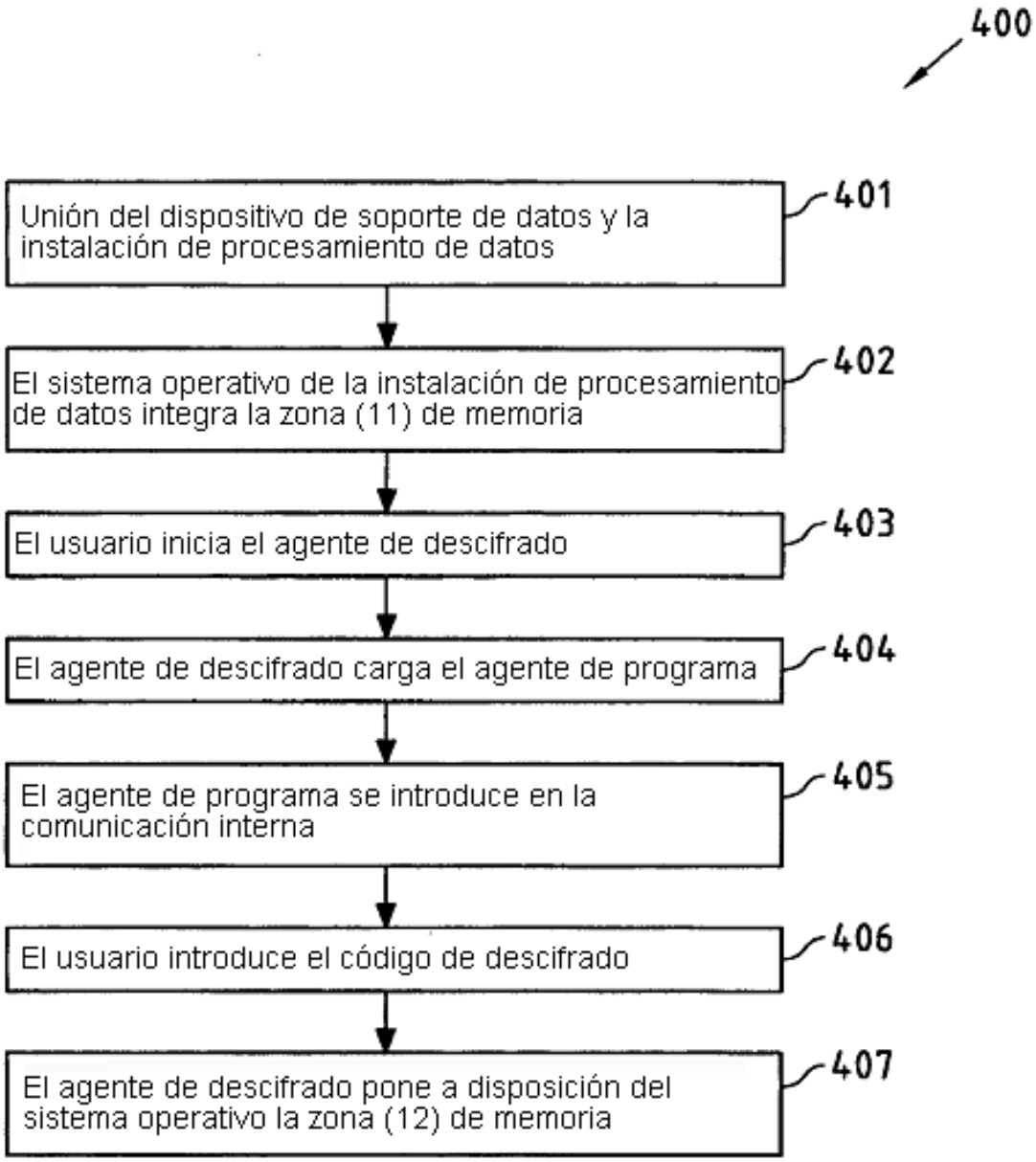
400
Unión del dispositivo de soporte de datos y la instalación de procesamiento de datos
401
El sistema operativo de la instalación de procesamiento de datos integra la zona (11) de memoria
402
El usuario inicia el agente de descifrado
403
El agente de descifrado carga el agente de programa
404
El agente de programa se introduce en la comunicación interna
405
El usuario introduce el código de descifrado
406
El agente de descifrado pone a disposición del sistema operativo la zona (12) de memoria
407

Fig.4a

Directriz de acción mínima:

4b

	Leer datos	Copiar datos	Imprimir datos	Transferir datos a través de red	Transferir datos a través de VPN	Borrar datos
Archivo 12b-1	+	–	–	–	–	–
Archivo 12b-2	+	–	–	–	–	–
Archivo 12b-3	+	–	–	–	–	–
Archivo 12b-4	+	–	–	–	–	–
Archivo 12b-5	+	–	–	–	–	–
Archivo 12b-6	+	–	–	–	–	–
Archivo 12b-7	+	–	–	–	–	–
Archivo 12b-8	+	–	–	–	–	–
Archivo 12b-9	+	–	–	–	–	–
Archivo 12b-10	+	–	–	–	–	–
Archivo 12b-11	+	–	–	–	–	–
Archivo 12b-12	+	–	–	–	–	–
Archivo 12b-13	+	–	–	–	–	–
Archivo 12b-14	+	–	–	–	–	–
Archivo 12b-15	+	–	–	–	–	–

+ : La acción con datos no incumple la directriz de acción

– : La acción con datos incumple la directriz de acción

Fig.4b

Directriz de acción media:

4c

	Leer datos	Copiar datos	Imprimir datos	Transferir datos a través de red	Transferir datos a través de VPN	Borrar datos
Archivo 12b-1	+	+	+	–	+	–
Archivo 12b-2	+	+	+	–	+	–
Archivo 12b-3	+	+	+	–	+	–
Archivo 12b-4	+	–	+	–	+	–
Archivo 12b-5	+	–	+	–	+	–
Archivo 12b-6	+	–	+	–	+	–
Archivo 12b-7	+	–	+	–	+	–
Archivo 12b-8	+	–	+	–	+	–
Archivo 12b-9	+	–	+	–	+	–
Archivo 12b-10	+	–	+	–	+	–
Archivo 12b-11	+	–	+	–	+	–
Archivo 12b-12	+	–	+	–	+	–
Archivo 12b-13	+	–	+	–	+	–
Archivo 12b-14	+	–	+	–	+	–
Archivo 12b-15	+	–	+	–	+	–

+ : La acción con datos no incumple la directriz de acción

– : La acción con datos incumple la directriz de acción

Fig.4c

Directriz de acción máxima:

4d

	Leer datos	Copiar datos	Imprimir datos	Transferir datos a través de red	Transferir datos a través de VPN	Borrar datos
Archivo 12b-1	+	+	+	+	+	+
Archivo 12b-2	+	+	+	+	+	+
Archivo 12b-3	+	+	+	+	+	+
Archivo 12b-4	+	+	+	+	+	+
Archivo 12b-5	+	+	+	+	+	+
Archivo 12b-6	+	+	+	+	+	+
Archivo 12b-7	+	+	+	+	+	+
Archivo 12b-8	+	+	+	+	+	+
Archivo 12b-9	+	+	+	+	+	+
Archivo 12b-10	+	+	+	+	+	+
Archivo 12b-11	+	+	+	+	+	+
Archivo 12b-12	+	+	+	+	+	+
Archivo 12b-13	+	+	+	+	+	+
Archivo 12b-14	+	+	+	+	+	+
Archivo 12b-15	+	+	+	+	+	+

+ : La acción con datos no incumple la directriz de acción

– : La acción con datos incumple la directriz de acción

Fig.4d

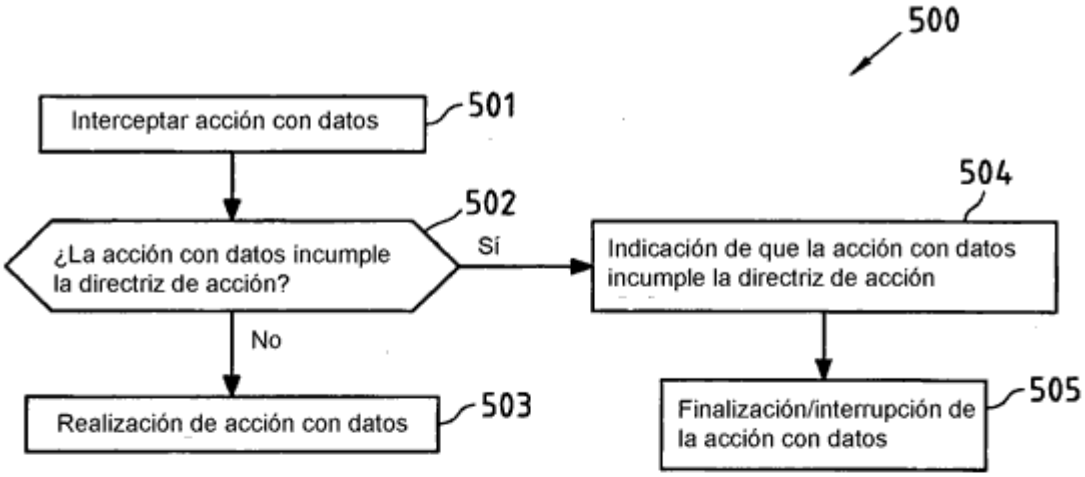
500
Interceptar acción con datos
501
502
¿La acción con datos incumple la directriz de acción?
Sí
504
Indicación de que la acción con datos incumple la directriz de acción
No
503
Realización de acción con datos
505
Finalización/interrupción de la acción con datos

Fig.5a

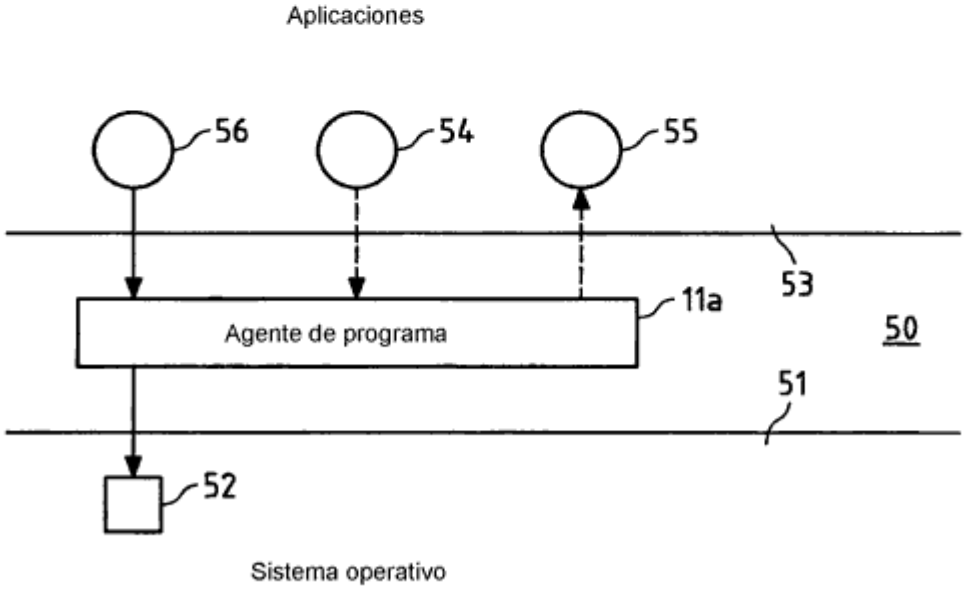
Aplicaciones
56
54
55
53
11a
Agente de programa
50
51
52
Sistema operativo

Fig.5b

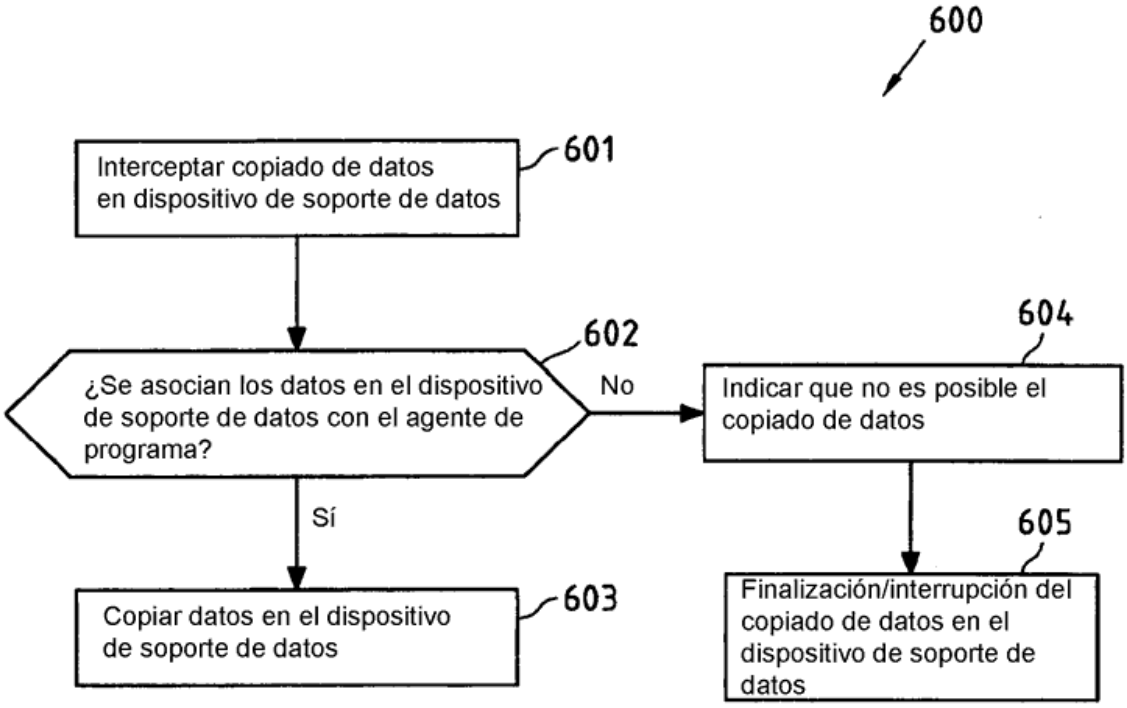
600
Interceptar copiado de datos en dispositivo de soporte de datos
601
¿Se asocian los datos en el dispositivo de soporte de datos con el agente de programa?
602
No
Sí
Indicar que no es posible el copiado de datos
604
Finalización/interrupción del copiado de datos en el dispositivo de soporte de datos
605
Copiar datos en el dispositivo de soporte de datos
603

Fig.6

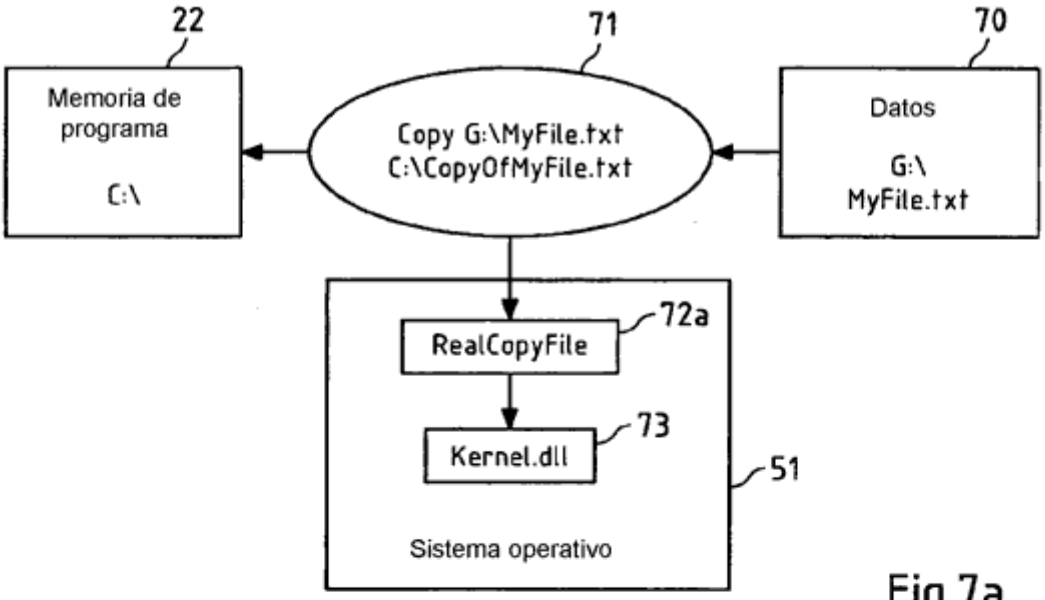
22
71
70
Memoria de programa
C:\
Copy G:\MyFile.txt C:\CopyOfMyFile.txt
Datos
G:\ MyFile.txt
72a
RealCopyFile
73
Kernel.dll
51
Sistema operativo

Fig.7a

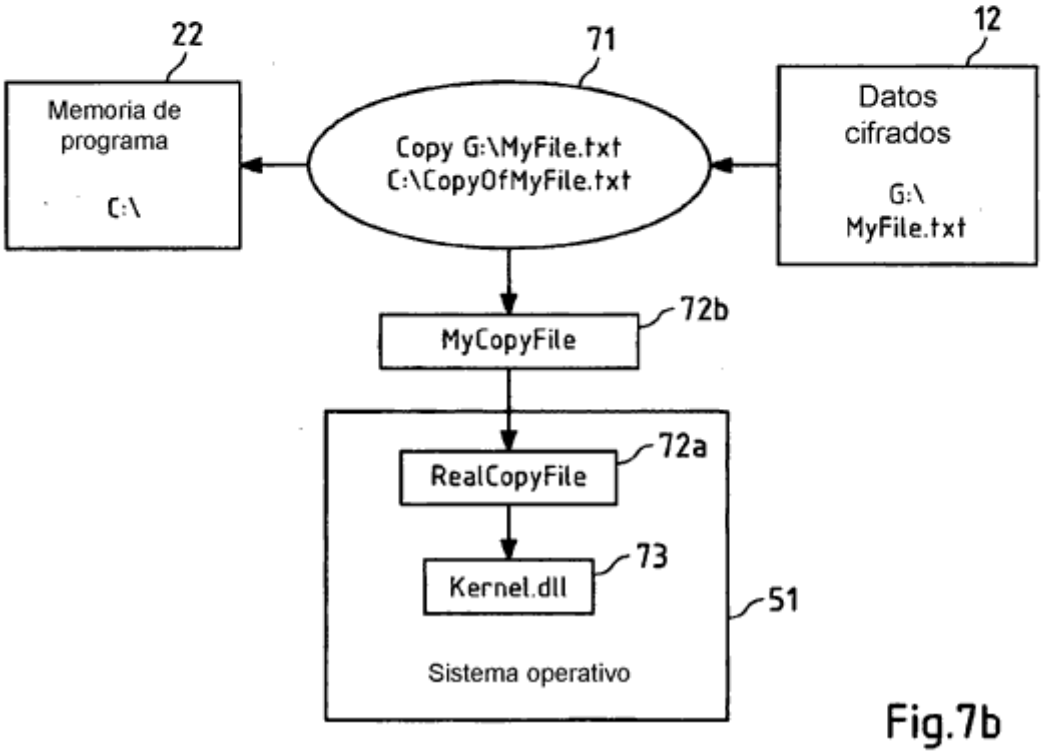
22
71
12
Memoria de programa
C:\
Copy G:\MyFile.txt C:\CopyOfMyFile.txt
Datos cifrados
G:\ MyFile.txt
72b
MyCopyFile
72a
RealCopyFile
73
Kernel.dll
51
Sistema operativo

Fig.7b

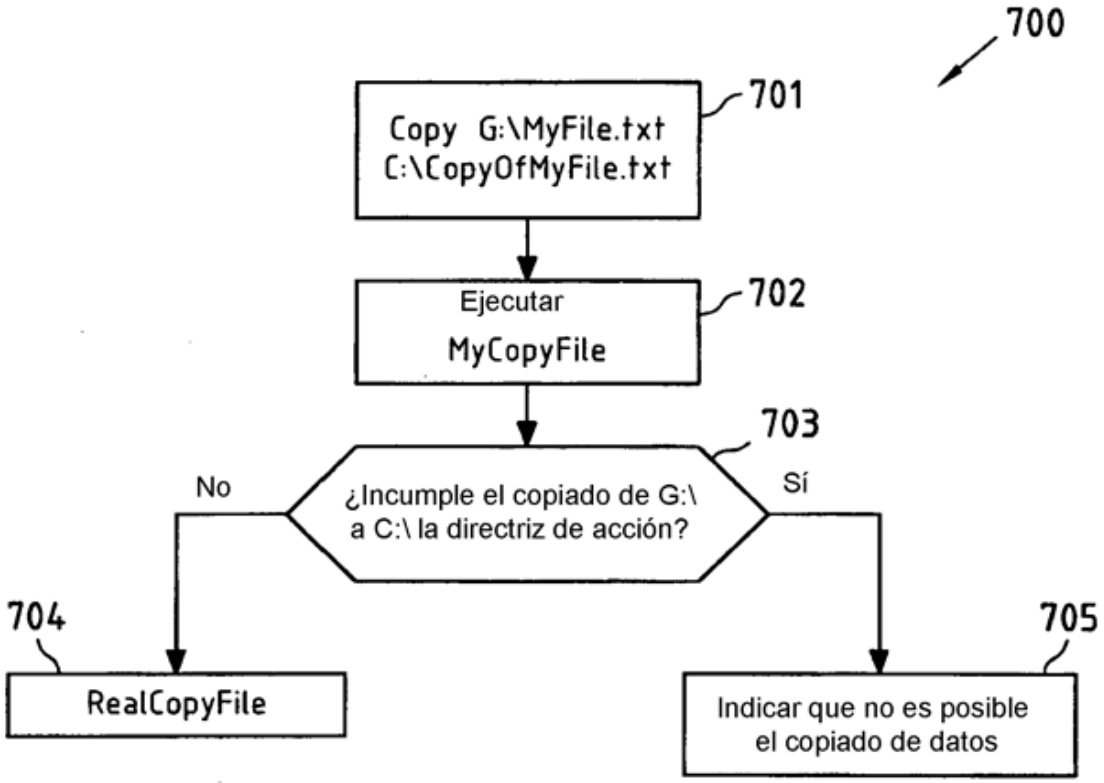

Fig.7c

```
MyCopyFile (origen, destino)
510 Si (CurrentProfile.ExportFile==true)
520 RealCopyFile (origen, destino)
530 si no
540 DisplayErrorMessage
550 fin si
```

Fig.7d