(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 650 448**

(51) Int. CI.:

H04L 29/06 (2006.01)
G06F 17/30 (2006.01)
G06F 21/57 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **08.08.2013 PCT/EP2013/066648**

(87) Fecha y número de publicación internacional: **03.04.2014 WO14048630**

(96) Fecha de presentación y número de la solicitud europea: **08.08.2013 E 13750290 (2)**

(97) Fecha y número de publicación de la concesión europea: **27.09.2017 EP 2870565**

(54) Título: **Comprobación de identidad de datos de propiedades de un aparato mediante un aparato de prueba**

(30) Prioridad:

28.09.2012 DE 102012217743

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
18.01.2018

(73) Titular/es:

SIEMENS AKTIENGESELLSCHAFT (100.0%)
Wittelsbacherplatz 2
80333 München, DE

(72) Inventor/es:

BUSSER, JENS-UWE y
FISCHER, KAI

(74) Agente/Representante:

LOZANO GANDIA, José

COMPROBACIÓN DE IDENTIDAD DE DATOS DE PROPIEDADES DE UN APARATO MEDIANTE UN APARATO DE PRUEBA

DESCRIPCIÓN

La invención se refiere a un procedimiento y a un sistema de prueba para una comprobación de identidad de datos de propiedades de un aparato mediante un aparato de prueba dentro de una red.

La inclusión de aparatos de medida o de control distribuidos en el entorno de la automatización de instalaciones tiene una importancia creciente. Al respecto se necesitan por ejemplo aparatos de campo inteligentes para la realización de redes eléctricas inteligentes, así como aparatos con función de control dentro de instalaciones de automatización o contadores para facturar servicios obtenidos, como comunicación, electricidad, gas o agua.

Al respecto se transmiten a menudo dentro de una red datos relevantes para la seguridad, como por ejemplo datos de medidas u órdenes de maniobra reunidos.

Las manipulaciones de tales aparatos significan un problema a tomar en serio para lograr un funcionamiento seguro, fiable y económico de infraestructuras, como por ejemplo instalaciones de automatización o redes de suministro de energía inteligentes. Bajo una manipulación ha de entenderse aquí una modificación no autorizada de datos en aparatos tales como aparatos de medida o de control. Ello incluye por ejemplo una modificación de las propiedades o funcionalidades de aparatos, como por ejemplo la sustitución de un software de aparatos (firmware) o la modificación de zonas individuales del software de aparatos. Además han de protegerse datos de configuración de un aparato frente a una modificación no intencionada y no autorizada.

La entidad solicitante conoce por el documento con el número de registro oficial de la Oficina Alemana de Patentes y Marcas 10 2011 077 289.8 un sistema y un procedimiento para detectar una variación del estado en una instalación de automatización. Allí se consulta a intervalos regulares la integridad del software de los aparatos o de los datos de configuración mediante un protocolo challenge-response (reto-respuesta). Si no coincide una huella dactilar calculada por un aparato con la huella dactilar esperada en un aparato de vigilancia, entonces se ha modificado el aparato sin autorización.

Por el documento de patente DE 10 2007 034 525 B4 se conoce un procedimiento que comprueba en una zona de memoria predeterminada datos memorizados en aparatos, comparando una huella dactilar calculada por un aparato con una huella dactilar calculada por una instancia de prueba.

La solicitud internacional WO 00/18162 da a conocer un procedimiento challenge-response en el que un retador (challenger) debe identificar o autentificar aparatos móviles o bien “user terminals” (terminales de usuarios) o bien “remote terminals” (terminales remotos). El retador es por ejemplo una “updating authority” (autoridad de actualización) y proporciona una actualización del software (software-update) en función de la comprobación del “remote terminal”. Entonces se introducen para una comprobación en una función hash el software de un aparato y un identificador, así como un reto (challenge) del retador (challenger). Este valor hash se genera en ambos lados, challenger y responder (contestador) y se comprueba la coincidencia. El documento tiene como base el problema de realizar una autentificación, por ejemplo de software, a través de un canal no protegido.

La solicitud de patente europea describe una prueba de integridad de una “computing platform” (plataforma de cálculo) mediante un procedimiento challenge-response con la formación de un valor hash mediante datos de prueba de la “computing platform”.

Para una gran cantidad de aparatos a comprobar dentro de una red, se espera de una instancia de prueba una gran potencia de cálculo, en particular cuando se realiza una comprobación en paralelo o prácticamente simultánea. A la vez debe quedar asegurado mediante la comprobación que se descubran las manipulaciones.

Partiendo de esta base, consiste el objetivo de la presente invención en proporcionar un procedimiento y un sistema de prueba que posibiliten una comprobación simplificada y segura frente a manipulaciones de la integridad del correspondiente aparato, cuando hay una gran cantidad de aparatos, mediante una instancia de prueba.

Este objetivo se logra mediante un procedimiento y un sistema de prueba según las características indicadas en las reivindicaciones independientes. Ventajosas formas de realización y perfeccionamientos se indican en las reivindicaciones secundarias.

Las ventajas citadas a continuación no tienen que lograrse necesariamente mediante los objetos de las reivindicaciones independientes. Más bien puede tratarse aquí también de ventajas que solamente se logran mediante formas de realización o perfeccionamientos individuales.

De acuerdo con la invención, se presenta un procedimiento para comprobar una integridad de datos de propiedades de un aparato (incluyendo los datos de propiedades un bloque de datos de programa y un bloque de datos de configuración) mediante un aparato de prueba dentro de una red, que incluye el aparato, el aparato de prueba y al menos otro aparato con otros datos de propiedades (incluyendo los otros datos de propiedades otro bloque de datos de programa y otro bloque de datos de configuración), las siguientes etapas:

Una estación, que está acoplada al menos con el aparato de prueba mediante la red, proporciona los datos de propiedades y los otros datos de propiedades. El aparato de prueba evalúa los datos de propiedades y los otros datos de propiedades en cuanto al menos una identidad del bloque de datos de programa y del otro bloque de datos de programa y asigna un identificador en función de un resultado de la evaluación.

El aparato de prueba determina un parámetro, que es adecuado para influir sobre un valor hash y un valor de comprobación hash de una función hash, estando configurado el parámetro en función del identificador y transmite el parámetro al aparato.

El aparato calcula el valor hash en dependencia de la función hash y de una cadena de caracteres, incluyendo la cadena de caracteres el parámetro y los datos de propiedades, pudiendo predeterminarse una secuencia dentro de la cadena de caracteres y transmite el valor hash al aparato de prueba.

El aparato de prueba calcula el valor hash de comprobación en dependencia de la función hash y de la cadena de caracteres, incluyendo la cadena de caracteres el parámetro y los datos de propiedades, pudiendo predeterminarse una secuencia dentro de la cadena de caracteres.

El aparato de prueba confirma la integridad de los datos de propiedades, en el caso de que el valor hash coincida con el valor hash de comprobación.

Como integridad de datos de propiedades de un aparato se entiende en la presente solicitud un estado que puede deducir que existe una manipulación del aparato o de sus datos de propiedades. Esto significa por ejemplo que los datos de propiedades en una comprobación no se han modificado ni sustituidos en comparación con los de una comprobación anterior de los datos de propiedades o que corresponden a un estado esperado, determinado o memorizado.

Como bloque de datos de programa de los datos de propiedades se entiende usualmente el software de un aparato o firmware de un aparato o datos de programa en forma de software. Un bloque de datos de configuración contiene usualmente parámetros como direcciones de red, identificadores de aparatos o material de claves criptográficas y certificados, así como parámetros que configuran la funcionalidad del software; los mismos son individuales para un aparato.

El procedimiento propuesto para comprobar un aparato dentro de una red posibilita considerar uno u otros varios aparatos dentro de la red, evaluando el aparato de prueba tanto los datos de propiedades del aparato como también los de los otros aparatos e iniciando las otras etapas del proceso de la comprobación en función del resultado de la valuación. Así se logra una posibilidad de comprobar un aparato dentro de una red en cuanto a manipulaciones, obteniéndose al tener en cuenta otros aparatos dentro de la red las premisas para una comprobación eficiente de una gran cantidad de aparatos. Así se detectan manipulaciones tempranamente.

Un aparato a comprobar determina entonces por ejemplo el valor hash de nuevo, es decir, el cálculo del valor hash se realiza en cada comprobación de nuevo con el parámetro prescrito de nuevo para cada comprobación.

Igualmente calcula el aparato de prueba el valor de comprobación hash para el aparato a comprobar en cada comprobación. Los datos de propiedades, como en particular el bloque de datos de proceso, así como el bloque de datos de configuración, los obtiene el aparato de prueba entonces de una estación, que tiene por ejemplo, la función de una estación de ingeniería dentro de una instalación de automatización industrial, en un estado actualizado en el momento de la comprobación. La estación es entonces por lo general parte integrante de la red y proporciona entonces tanto los datos de configuración de los aparatos a comprobar como también las versiones de firmware y software instaladas en los aparatos.

La comprobación de aparatos existentes en una instalación mediante un servidor de pruebas común, que está optimizado para la comprobación de muchos aparatos en paralelo, reduce un sobrecoste que se produciría debido a sistemas de prueba individuales para uno para pocos aparatos.

Según una forma de realización, se transmite el parámetro al comienzo de la cadena de caracteres.

Así no le resulta posible al aparato a comprobar memorizar para el cálculo del valor hash un estado interior del procedimiento hash, que se calcula sólo en función del bloque de datos de programa y del

bloque de datos de configuración, a continuación modificar el bloque de datos de programa y/o el bloque de datos de configuración y de nuevo, a continuación, calcular no obstante un valor hash correcto, que coincida con el valor hash de comprobación. Esto sería fácilmente posible en el caso de que el parámetro se transfiera al final de la cadena de caracteres a los que aplicar hash.

Así debe calcular el aparato a comprobar en una comprobación el valor hash en función del parámetro transmitido para la comprobación actual, así como el bloque de datos de programa y el bloque de datos de configuración que se encuentran en ese momento en el aparato.

Según una forma de realización, está configurado el parámetro como un número aleatorio, cuya longitud se adapta al valor hash.

El número aleatorio es entonces usualmente un número de la longitud 128 bits a 512 bits (16 a 128 bytes). La longitud debe corresponder entonces idealmente a la longitud del valor hash del procedimiento hash criptográfico utilizado. Como algoritmo hash de seguridad (Secure Hash Algorithm SHA) procede entonces en particular SHA-256, SHA-384 o SHA-512. Si se acorta el valor hash determinado antes de la transmisión de retorno, porque la cantidad de datos transmitidos que debe mantenerse reducida y también un valor hash más corto ofrece ya un escalón de seguridad suficiente, entonces puede configurarse también el número aleatorio correspondientemente corto.

Según una forma de realización, se transmiten el parámetro y/o el valor hash asegurados criptográficamente.

Si se realiza la transmisión del parámetro codificada, entonces se reduce el riesgo de que otros aparatos, a los cuales no está destinada la información sobre el parámetro, por ejemplo el número aleatorio, lleguen a conocer el número aleatorio. Sólo un aparato con la clave adecuada puede llegar a conocer el número aleatorio.

Una transmisión firmada del valor hash desde el aparato a comprobar al aparato de prueba garantiza la autenticidad de la respuesta del aparato a comprobar. Así puede detectar el aparato de prueba mediante una clave pública que el valor hash no se transmite desde el aparato a comprobar. Esto es en particular procedente para detectar y/o evitar ataques Denial-of-Service (de denegación de servicio) mediante otros aparatos en la red.

Puede pensarse, como alternativa a la firma digital, en utilizar un procedimiento hash keyed (codificado), en el que tanto por parte del aparato a comprobar como también por parte del aparato de prueba se calcula mediante una clave secreta un Message Authentication Code (MAC, código de autentificación de mensajes para el mensaje transmitido) y debe coincidir el correspondiente MAC calculado.

Una transmisión codificada del valor hash desde el aparato a comprobar al aparato de prueba impide además que otros aparatos de la red, que han recibido el mismo número aleatorio y que deben poseer el mismo bloque de datos de programa y de configuración, cuando ya no posean los mismos debido a una manipulación y por lo tanto ya no puedan calcular por sí mismos el valor hash correcto, simplemente intercepten la transmisión del valor hash y envíen por sí mismos este valor hash a continuación al aparato de pruebas, para engañar al aparato de pruebas.

Según un perfeccionamiento, se transmite un parámetro idéntico al aparato y al menos a otro aparato.

Precisamente en una comprobación en paralelo de muchos aparatos, simplifica la comprobación un parámetro idéntico, por ejemplo un número aleatorio, que se envía a todos los aparatos a comprobar, ya que el coste en cálculo en el aparato de prueba se mantiene reducido.

Según un perfeccionamiento, calcula el aparato de prueba, para el cálculo del valor hash de comprobación, en función del identificador, un estado interno en dependencia de la función hash, del parámetro y del bloque de datos de programa y memoriza el estado interno.

El tamaño del bloque de datos de programa depende fuertemente del aparato y puede llegar para aparatos a comprobar desde algunos KB, por ejemplo en el caso de un tag RFID, hasta varias docenas de MB, por ejemplo en el caso de aparatos de campo inteligentes o de controles programables en memoria (PLCs). El tamaño del bloque de datos de configuración es por el contrario usualmente bastante menor. El mismo se encuentra la mayoría de las veces en la gama de como máximo unos pocos tantos por ciento del bloque de datos de programa. El tamaño del parámetro es frente al bloque de datos de programa tan pequeño que puede despreciarse.

Con ello origina el bloque de datos de programa el máximo coste de cálculo al calcular el valor hash de comprobación.

Dentro de una instalación existen por lo general relativamente pocos bloques de datos de programa diferentes, ya que los aparatos de una serie constructiva tienen la mayoría de las veces todos el mismo

firmware y/o software o como máximo existen pocas versiones diferentes dentro de una instalación. Con ello la cantidad de bloques de datos de programa diferentes utilizados en una instalación está fuertemente limitado, incluso cuando estén instalados muchos aparatos en la instalación.

Si ahora detecta el aparato de prueba al evaluar los datos de propiedades y los otros datos de propiedades que los correspondientes bloques de datos de programa son idénticos, entonces puede calcular el aparato de prueba, al enviar un parámetro idéntico a los aparatos con idéntico bloque de datos de programa, un estado interno válido conjuntamente para aparatos que tienen un bloque de datos de programa común. Esto significa para el servidor de prueba una reducción del coste de cálculo, ya que el mismo tiene que calcular el valor hash de comprobación sólo tras calcular el estado interno, en cada caso en función del estado interno y del bloque de datos de configuración, que ahora se calcula individualmente para cada aparato.

Según otra forma de realización, transmite el aparato de prueba un parámetro único al aparato y otro parámetro único distinto a al menos otro aparato, transmitiéndose el parámetro único enviado y el otro parámetro único enviado dentro de una ventana de tiempo que puede predeterminarse.

Una ventaja de esta comprobación casi simultánea de muchos aparatos a comprobar con distintos parámetros, como en particular números aleatorios, consiste en que se obstaculizan ataques de retransmisión. Un aparato instalado en la instalación para un ataque de retransmisión, al que se deriva el cálculo de un valor hash, para ocultar así una manipulación del aparato a comprobar, tiene ahora un coste de comprobación extremadamente elevado: Se recibe ahora de cada aparato manipulado, cuya solicitud de cálculo del valor hash se retransmite, un parámetro diferente para calcular el valor hash. Por ejemplo puede descubrirse un ataque de retransmisión mediante tiempos de respuesta inesperadamente elevados de los aparatos comprobados.

Según una forma de realización, transmite el aparato de pruebas un parámetro único al aparato y otro parámetro único a al menos otro aparato en función del identificador.

Con ello es posible elegir en aparatos con el mismo bloque de datos de programa un parámetro a partir de un pequeño número de parámetros distintos, con lo que para aparatos que tienen el mismo bloque de datos de programa no puede partirse de un parámetro idéntico. Debido a ello ciertamente aumenta el coste de cálculo para el aparato de prueba en comparación con un parámetro idéntico para todos los aparatos de una instalación, pero desde luego así es más difícil para aparatos manipulados intercambiar el valor hash correcto. Si por ejemplo varios aparatos manipulados han memorizado conjuntamente todas las partes de un bloque de datos de programa original, entonces pueden confeccionar los mismos en cooperación el valor hash correcto. Desde luego si reciben los mismos distintos parámetros, se dificulta esta sintonización y aumenta el coste de cálculo, ya que ahora para cada parámetro que ha recibido uno de los aparatos debe calcularse otro valor hash.

Según otro perfeccionamiento, se confirma la integridad del aparato mediante el aparato de prueba cuando el cálculo y transmisión del valor hash se realiza mediante el aparato dentro de una ventana de tiempo de respuesta que puede predeterminarse de forma individual.

En el caso de un ataque de retransmisión, es entonces prácticamente imposible que varios aparatos manipulados contesten dentro de la ventana de tiempo de respuesta predeterminada mediante la transmisión del valor hash, ya que un aparato al que se desvía el cálculo del valor hash tiene que invertir un coste de cálculo demasiado alto para responder en el tiempo predeterminado. En particular, en el caso de que el aparato de pruebas inicie simultáneamente el procedimiento de comprobación de muchos aparatos, puede descubrirse así un ataque de retransmisión.

Según un perfeccionamiento, calcula el aparato de pruebas el valor hash de comprobación antes o después de la transmisión del parámetro único.

En el caso de un cálculo separado del valor hash de comprobación para cada aparato, se genera para el aparato de pruebas un elevado coste de cálculo, por lo que es ventajoso ejecutar los cálculos de los valores de prueba hash ya antes de enviar el parámetro, con lo que el aparato de pruebas inmediatamente después de recibir el valor hash de un aparato a comprobar, puede comprobar rápidamente que es correcto y dado el caso puede activar una alarma. Un cálculo del valor hash de comprobación mediante el aparato de prueba tras recibir el valor hash como respuesta del aparato a comprobar es igualmente posible, detectándose por supuesto sólo posteriormente un valor hash falso y activándose sólo posteriormente dado el caso una alarma. Cuando existe un aviso de advertencia o avisos relativos a la manipulación de un aparato o de un grupo de aparatos, puede no obstante elegirse este procedimiento, para asegurar una elevada seguridad en la comprobación.

La invención incluye además un sistema de prueba para comprobar una integridad de datos de propiedades de un aparato, incluyendo los datos de propiedades un bloque de datos de programa y un bloque de datos de configuración, mediante un aparato de pruebas dentro de una red, que incluye el aparato, el aparato de pruebas y al menos otro aparato con otros datos de propiedades, incluyendo los

otros datos de propiedades otro bloque de datos de programa y otro bloque de datos de configuración, incluyendo:

- una estación para proporcionar los datos de propiedades y los otros datos de propiedades, estando acoplada la estación al menos con el aparato de prueba mediante la red;
- el aparato de prueba para evaluar los datos de propiedades y los otros datos de propiedades en cuanto a al menos una identidad del bloque de datos de programa y del otro bloque de datos de programa, asignación de un identificador en función de un resultado de la evaluación, determinación de un parámetro, que es adecuado para influir sobre un valor hash y un valor hash de comprobación de una función hash, configurándose el parámetro en función del identificador, transmisión del parámetro al aparato, cálculo del valor hash de comprobación en dependencia de la función hash y de una cadena de caracteres, incluyendo la cadena de caracteres el parámetro y los datos de propiedades y pudiendo predeterminarse una secuencia dentro de la cadena de caracteres y confirmación de la integridad de los datos de propiedades, en el caso de que el valor hash coincida con el valor hash de comprobación;
- el aparato para calcular el valor hash en dependencia de la función hash y de una cadena de caracteres, incluyendo la cadena de caracteres el parámetro y los datos de propiedades y pudiendo predeterminarse una secuencia dentro de la cadena de caracteres y transmisión del valor hash al aparato de prueba.

Según una forma de realización incluye el sistema de prueba además al menos otra unidad adicional para utilizarla en una de las etapas del procedimiento según las formas de realización del procedimiento antes citadas.

Según otra forma de realización, están configurados el aparato de pruebas y la estación como unidad común.

Con ello no tiene que intercambiarse ningún dato a través de la red, como en particular los a menudo voluminosos datos de propiedades.

La invención se describirá a continuación más en detalle con ejemplos de realización en base a las figuras. Se muestra en:

figura 1 una representación esquemática del procedimiento de comprobación según un primer ejemplo de realización de la invención;
figura 2 una representación esquemática del procedimiento de comprobación según un segundo ejemplo de realización de la invención;
figura 3 una representación esquemática del procedimiento de comprobación según un tercer ejemplo de realización de la invención.

En las figuras se han dotado los mismos elementos o elementos que funcionan de la misma manera de las mismas referencias, siempre que no se indique otra cosa.

En base a la figura 1 se presentarán las etapas básicas del procedimiento y los componentes esenciales del procedimiento de acuerdo con la invención.

Dentro de una instalación de automatización existen numerosos aparatos, como aparatos de medida o control, que han de comprobarse para lograr un funcionamiento seguro y fiable de la instalación en cuanto a manipulaciones. Para comprobar un aparato 100, que por ejemplo es un control de robot correspondiente a un robot industrial, se propone para ello aportar un aparato de prueba 300, que a través de una red 400 de la instalación de automatización está acoplado con el aparato 100, de una estación 200, que está acoplada a través de la red 400 al menos con el aparato de prueba 300, para aportar datos de propiedades ED1 del aparato 100. Los datos de propiedades ED1 hacen posible una identificación inequívoca del aparato 100, es decir, del control de robot. La estación está configurada entonces como un servidor de actualización (update), que aporta al aparato de pruebas 300 continuamente una versión actualizada de los datos de propiedades ED1. También se proporcionan otros datos de propiedades ED2 de otro aparato 102 dentro de la instalación al aparato de pruebas 300 desde la estación 200.

Se realiza entonces una evaluación de los datos de propiedades ED1 y de los otros datos de propiedades ED2 proporcionados al aparato de prueba 300. Los datos de propiedades ED1, así como los otros datos de propiedades ED2, incluyen un bloque de datos de programa PD1, así como un bloque de datos de configuración KD1 y/u otro bloque de datos de programa PD2, así como otro bloque de datos de configuración KD2 del otro aparato 102. El bloque de datos de programa PD1 contiene una indicación sobre el firmware que se encuentra en el control de robot. El bloque de datos de configuración contiene un identificador de aparato y parámetros que han sido prescritos para la secuencia de etapas del proceso del robot industrial.

Los correspondientes datos de propiedades, que proporciona la estación 200 al aparato de pruebas 300, en este caso el servidor de update, se evalúan ahora en cuanto al menos una identidad de los

correspondientes bloques de datos de programa. En función del resultado de la evaluación, se asigna a un bloque de datos de programa de un aparato a comprobar, en el ejemplo descrito el bloque de datos de programa PD1 del aparato 100, un identificador Z. El identificador Z de bloques de datos de programa, en los que se ha detectado entre sí una identidad, se realiza entonces agrupando lógicamente el aparato de pruebas 300 los datos de propiedades que se le aportan, por ejemplo datos de propiedades que se le aportan con datos de propiedades proporcionados con idéntico bloque de datos de programa en una zona de memoria común. Los controles de robot de una serie constructiva común con números de versión idénticos pueden así reunirse lógicamente mediante el aparato de pruebas 300.

Ahora se determina en base a este identificador Z mediante el aparato de prueba 300 un parámetro P, que por un lado se transmite al aparato 100 para calcular un valor hash H1 mediante el aparato 100 y por otro lado se utiliza para calcular un valor hash de comprobación H'1 mediante el aparato de pruebas 300. El cálculo del valor hash H1 por parte del aparato 100, así como del valor hash de comprobación H'1 por parte del aparato de pruebas 300 se realiza en cada caso en dependencia de una función hash y de una cadena de caracteres. La cadena de caracteres incluye entonces el parámetro P y los datos de propiedades ED1, aplicándose el hash a la cadena de caracteres en cada caso en una forma tal que el parámetro P se transfiere como primero y a continuación el bloque de datos de programa PD1, así como el bloque de datos de configuración KD1.

El aparato 100 puede calcular entonces un valor hash correcto, que coincide con el valor hash de comprobación H'1 calculado por el aparato de prueba 300 sólo cuando realiza el cálculo según la solicitud de cálculo del valor hash H1 - realizándose esto mediante la transmisión del parámetro P al aparato 100 - con los datos de propiedades ED1 que se encuentran en ese momento en el aparato 100. Un estado interno del procedimiento hash en función de los datos de propiedades ED1 no puede así calcularse previamente y memorizarse de manera razonable. Cuando se introduce en el cálculo un parámetro P calculado como nuevo para cada procedimiento de prueba, por ejemplo un número aleatorio y se transfiere al comienzo de la cadena de caracteres, no puede calcularse con datos de propiedades ED1 no manipulados inicialmente un estado interno del procedimiento hash, que incluso después de la manipulación de los datos de propiedades ED1 suministrase aún un valor hash correcto, tal como sería posible en un procedimiento en el que el parámetro P se encontrase al final de la cadena de caracteres.

Una inclusión de los otros aparatos dentro de la instalación al determinar el parámetro P, implica para el aparato de prueba una reducción del coste de cálculo, tal como se describirá más en detalle a continuación en base a la figura 2.

Se describirá un ejemplo de realización en base a tres aparatos a comprobar. Análogamente al caso antes descrito, proporciona la estación 200 datos de propiedades, es decir, en este caso los datos de propiedades ED1, los otros datos de propiedades ED2, así como terceros datos de propiedades ED3. Los datos de propiedades ED1 incluyen el bloque de datos de programa PD1, así como el bloque de datos de configuración KD1 y los otros datos de propiedades ED2 incluyen igualmente el bloque de datos de programa PD1, que es idéntico al bloque de datos de programa PD1 del aparato 100, así como el otro bloque de datos de configuración KD2 y los terceros datos de propiedades ED3 incluyen un tercer bloque de datos de programa PD3, así como un tercer bloque de datos de configuración KD3. El aparato de prueba 300 toma ahora una identificación de los datos de propiedades ED1, así como de los otros datos de propiedades ED2, ya que la evaluación de los correspondientes datos de propiedades ha dado como resultado la identidad de los respectivos bloques de datos de programa.

Al aparato 100 y al otro aparato 102 se les transmite ahora un parámetro idéntico PC, en base a los bloques de datos de programa idénticos. Si para el tercer aparato 103 no se detectó ninguna identidad del tercer bloque de datos de programa PD3 con uno de los otros bloques de datos de programa de los aparatos a comprobar, entonces se transmite al tercer aparato 103 un parámetro cualquiera P. El aparato 100, el otro aparato 102 y el tercer aparato 103 calculan, análogamente al procedimiento antes descrito, el valor hash H1, otro valor hash H2 y un tercer valor hash H3. Se realiza en cada caso la transmisión de los valores hash calculados desde los correspondientes aparatos al aparato de prueba 300.

El aparato de prueba 300 puede calcular el correspondiente valor de prueba hash ahora según un procedimiento simplificado. Para el bloque de datos de programa PD1 que se le transmite, calcula el aparato de pruebas 300 ahora primeramente un estado interno IZ, que depende de la función hash, del correspondiente parámetro PC y del bloque de datos de programa PD1. Este estado interno IZ se memoriza y se realiza a continuación el cálculo del valor hash de comprobación H'1, calculándose el valor hash de comprobación H'1 en dependencia de la función hash, del estado interno IZ y del bloque de datos de configuración KD1.

Para el cálculo del valor hash de comprobación H'2 del otro aparato 102, se utiliza igualmente el estado interno IZ, con lo que sólo es necesario realizar un hash del otro bloque de datos de configuración KD2.

Este procedimiento puede ampliarse a una cantidad de aparatos cualquiera, realizándose un cálculo y memorización del estado interno IZ siempre cada vez que se transmiten datos de programa. Esta síntesis del cálculo del estado interno se realiza independientemente de la cantidad de aparatos existentes.

Cuantos más aparatos se reúnan mediante este procedimiento, tanto mayor será la reducción del coste del cálculo.

Para grandes cantidades de aparatos, significa este procedimiento por lo tanto para el aparato de pruebas 300 una gran simplificación, lo cual se mostrará en base al siguiente ejemplo numérico.

Un tamaño de datos de programa de un aparato, como por ejemplo firmware o software, depende fuertemente del aparato y puede alcanzar desde varios kB (por ejemplo para tags RFID) hasta algunas docenas de MB (por ejemplo en el caso de aparatos de campo inteligentes o de controles programables en memoria). Para el ejemplo numérico se supondrá un tamaño medio de datos de programa GPD de GPD = 32 MB. Debe suponerse que dentro de una instalación existe una cantidad de datos de programa M de diversos bloques de datos de programa, en particular diversas series constructivas, con M = 10.

Un tamaño promedio de datos de configuración GKD es usualmente inferior al de los datos de programa y es en el ejemplo de GKD = 0,1 MB. Un número de aparatos diversos, por ejemplo un número de 1000 aparatos dentro de una instalación, significa una cantidad de datos de configuración N de N = 1000.

Para un método de comprobación sin utilizar un estado interno IZ para los aparatos a comprobar con idéntico bloque de datos de programa, se compone la cantidad de datos G1 a los que aplicar hash para el aparato de prueba 300 por:

$$N \cdot GPD + N \cdot GKD = G1$$

Esto corresponde al caso en el que para cada aparato de una instalación tenga que calcularse un valor hash y la cantidad de datos G1 a los que aplicar hash se compone como suma del producto de la cantidad de datos de configuración N y del tamaño promedio de datos de programa GPD y el producto de la cantidad de datos de configuración N y el tamaño promedio de datos de configuración GKD.

Cuando calcula el aparato de prueba 300 primeramente sólo para cada bloque de datos de programa diferente un estado interno IZ y a continuación el valor hash relativo a los datos de configuración, resulta entonces la siguiente cantidad de datos adicional G2 a la que aplicar hash:

$$M \cdot GPD + N \cdot GKD = G2$$

Para la variante sin utilizar el estado interno IZ, esto significa en el ejemplo numérico una cantidad de datos G1 a los que aplicar hash de:

$$1000 \cdot 32MB + 1000 \cdot 0{,}1MB = 32100MB$$

La otra cantidad de datos G2 a los que aplicar hash utilizando el estado interno IZ es por el contrario sólo de:

$$10 \cdot 32MB + 1000 \cdot 0{,}1MB = 420\ MB$$

La cantidad de datos a los que aplicar hash tiene entonces correlación con el correspondiente coste de cálculo, que en el caso del cálculo de un estado interno IZ es igualmente menor, debido a la inferior cantidad de datos a los que aplicar hash, con lo que la comprobación de los aparatos se realiza más rápidamente.

Para unas mayores exigencias de seguridad, se describirá en otro ejemplo de realización en base a la figura 3 cómo no obstante puede mantenerse reducido el coste en cálculo para un aparato de prueba 300. Al respecto se transmiten, en el caso de un identificador Z de datos de propiedades con idénticos bloques de datos de programa, parámetros en parte diferentes a los aparatos con datos de características identificados. En la figura 3 se representa cómo transmite un aparato 100 con datos de propiedades ED1 y un bloque de datos de programa PD1 un parámetro único PS1, transmitiéndose a otro aparato 102 con otros datos de propiedades ED2 y un bloque de datos de programa PD1 idéntico al bloque de datos de programa PD1 del aparato 100, otro parámetro único PS2. Para un tercer aparato 103 con terceros datos de propiedades ED3 y un tercer bloque de datos de programa PD3, que no coincide con uno de los bloques de datos de programa de los demás aparatos, puede transmitirse cualquier tercer parámetro único PS3, que puede coincidir con uno de los demás parámetros transmitidos. Entonces dispone el aparato de prueba 300 para elegir un parámetro único a enviar por ejemplo de algunos números aleatorios de un pequeño número de datos aleatorios L.

El coste en cálculo para el aparato de prueba según este método aumenta debido a que para cada bloque de datos de programa existente en la instalación sólo tiene que calcularse ahora un estado interno por cada posible número aleatorio. Resulta una tercera cantidad de datos G3 a la que aplicar hash:

$$L \cdot M \cdot GPD + N \cdot GKD = G3$$

En el ejemplo numérico antes citado resulta así, para elegir el número aleatorio entre L = 10 diversos números aleatorios, una tercera cantidad de datos G3 a los que aplicar hash de:

$$10 \cdot 10 \cdot 32MB + 1000 \cdot 0,1MB = 3300MB$$

Se trata al respecto de un procedimiento con elevada seguridad, ya que ahora se dificulta un acuerdo entre aparatos manipulados que realizan intercambios para averiguar un valor hash correcto. Por ejemplo pueden reconstruir varios aparatos manipulados conjuntamente un bloque de datos de programa original. Aún cuando el bloque de datos de programa se ha modificado para cada aparato, es posible no obstante calcular en cooperación un valor hash correcto. Desde luego cuando se transmiten distintos números aleatorios a los aparatos, se dificulta el que se pongan de acuerdo. Cada aparato debe calcular ahora, al suponerse que no es igual el número aleatorio, un valor hash correcto individualmente, teniendo que determinarse también el estado interno IZ correcto, ya individual, en particular en función del número aleatorio y del bloque de datos de programa.

Cuando la capacidad de cálculo disponible de una instancia de prueba es mayor, puede pensarse además en enviar a todos los aparatos a comprobar dentro de una instalación un parámetro P diferente. Este procedimiento puede elegirlo en particular el aparato de prueba 300 si se dispone de suficiente tiempo para calcular los valores hash de prueba o bien se tiene próximamente una comprobación general con exigencias de seguridad muy elevadas. Los parámetros P individuales para cada aparato a comprobar se transmiten entonces dentro de una ventana de tiempo que puede predeterminarse.

Una integridad de los datos de propiedades de un aparato a comprobar sólo se confirma en el caso de que tenga lugar una respuesta en forma del valor hash determinado y transmitido por el aparato a comprobar dentro de una ventana de tiempo de respuesta predeterminada.

Al respecto puede tener en cuenta el aparato de prueba que algunos aparatos a comprobar que tengan un elevado grado de ocupación en su procesador - por ejemplo debido a funciones voluminosas o a que exista una reducida capacidad de cálculo - necesitan un tiempo de respuesta correspondientemente más largo y puede bien adaptarse la ventana de tiempo de respuesta específicamente para el aparato o bien iniciarse correspondientemente más temprano una consulta de comprobación en forma de un parámetro transmitido para estos aparatos. Para el aparato de prueba 300 significa este procedimiento un coste de cálculo enormemente alto, por lo que es ventajoso realizar el cálculo de valores hash de prueba a calcular con antelación, es decir, antes del envío de los parámetros.

Los ataques de retransmisión dentro de una instalación se dificultan mediante este procedimiento, ya que se inicia una comprobación casi simultánea de todos los aparatos que se encuentran dentro de la instalación y cada aparato tiene que contestar dentro de una ventana de tiempo de respuesta predeterminada.

Los ejemplos de realización presentados pueden combinarse entre sí de manera ventajosa: Para ello pueden elegir el aparato de prueba como ajuste estándar el método en el que se reúna un cálculo de valores de prueba hash de aparatos con idénticos bloques de datos de programa, se envía a esos aparatos un parámetro PC idéntico y se calcula un estado interno IZ, con lo que el coste del cálculo para el aparato de prueba se mantiene bajo incluso cuando hay muchos aparatos a comprobar en paralelo. Cuando hay una sospecha de manipulación de un grupo de aparatos con idénticos bloques de datos de programa, puede recurrirse selectivamente al método en el que a aparatos con idénticos bloques de datos de programa se les transmitan distintos parámetros PS1, PS2. También cuando se instala un nuevo grupo de aparatos dentro de la instalación, puede ser ventajoso esto.

Así pueden cumplirse dentro de una instalación distintos niveles de seguridad para distintas exigencias de seguridad, variando el número de parámetros utilizados, en particular de los números aleatorios utilizados. También se detectan modificaciones de software no intencionadas, por ejemplo faltas de coincidencia entre el software y/o la configuración memorizado/a en la estación de ingeniería y el software y/o la configuración realmente existente en el aparato, debidas a faltas en la memoria o debidas a la configuración realizada localmente por un técnico de service y un ajuste incorrecto con la estación de ingeniería. La combinación de los distintos procedimientos presentados permite una comprobación eficiente en instalaciones con una gran cantidad de aparatos instalados.

Las unidades aparato 100, otro aparato 102, tercer aparato 103, aparato de prueba 300 y estación 200, así como la otra unidad, pueden realizarse y ejecutarse en software, hardware o en una combinación de software y hardware.

Así pueden estar archivadas las etapas realizadas por las unidades como código de programa sobre un medio de memoria, en particular un disco duro, CD-ROM o un módulo de memoria, diseñándose y procesándose las distintas instrucciones del código de programa mediante al menos una unidad de cálculo, que incluye un procesador. El procesador está conectado con el medio de memoria a través de un bus para el intercambio de datos.

Además puede conectarse una unidad de entrada/salida mediante el bus, pudiendo recibirse y/o enviarse mediante la unidad de entrada/salida datos, como por ejemplo datos de propiedades, un parámetro, un valor hash o un valor hash de comprobación.

Las formas de realización y perfeccionamientos descritos pueden combinarse libremente entre sí.

REIVINDICACIONES

1. Procedimiento para comprobar una integridad de datos de propiedades (ED1) de un aparato (100), en el que los datos de propiedades (ED1) incluyen un bloque de datos de programa (PD1) y un bloque de datos de configuración (KD1) mediante un aparato de prueba (300) dentro de una red (400), que incluye el aparato (100), el aparato de prueba (300) y al menos otro aparato (102) con otros datos de propiedades (ED2), incluyendo los otros datos de propiedades (ED2) otro bloque de datos de programa (PD2) y otro bloque de datos de configuración (KD2), con las siguientes etapas:
   - aportación de los datos de propiedades (ED1) y de los otros datos de propiedades (ED2) mediante una estación (200), que está acoplada al menos con el aparato de prueba (300) mediante la red (400);
   - evaluación de los datos de propiedades (ED1) y de los otros datos de propiedades (ED2) mediante el aparato de prueba (300) en cuanto al menos una identidad del bloque de datos de programa (PD1) y del otro bloque de datos de programa (PD2) y asignación de un identificador (Z) en función de un resultado de la evaluación;
   - determinación de un parámetro (P, PC, PS1), que es adecuado para influir sobre un valor hash (H1) y un valor de comprobación hash (H'1) de una función hash, mediante el aparato de prueba (300), estando configurado el parámetro (P, PC, PS1) en función del identificador (Z) y transmisión del parámetro (P, PC, PS1) al aparato (100);
   - cálculo del valor hash (H1) mediante el aparato (100) en dependencia de la función hash y de una cadena de caracteres, incluyendo la cadena de caracteres el parámetro (P, PC, PS1) y los datos de propiedades (ED1), pudiendo predeterminarse una secuencia dentro de la cadena de caracteres y transmitiéndose el valor hash (H1) al aparato de prueba (300);
   - cálculo del valor hash de comprobación (H'1) mediante el aparato de prueba (300) en dependencia de la función hash y de la cadena de caracteres, incluyendo la cadena de caracteres el parámetro (P, PC, PS1) y los datos de propiedades (ED1) y pudiendo predeterminarse una secuencia dentro de la cadena de caracteres;
   - confirmación de la integridad de los datos de propiedades (ED1) mediante el aparato de prueba (300), en el caso de que el valor hash (H1) coincida con el valor hash de comprobación (H'1).

2. Procedimiento de acuerdo con una de las reivindicaciones precedentes, en el que el parámetro (P, PC, PS1) se transfiere al comienzo de la cadena de caracteres.

3. Procedimiento de acuerdo con una de las reivindicaciones precedentes, en el que el parámetro (P, PC, PS1) está configurado como un número aleatorio, cuya longitud se adapta al valor hash (H1).

4. Procedimiento de acuerdo con una de las reivindicaciones precedentes, en el que el parámetro (P, PC, PS1) y/o el valor hash (H1) se transmiten asegurados criptográficamente.

5. Procedimiento de acuerdo con una de las reivindicaciones precedentes, en el que al aparato (100) y a al menos otro aparato (102) se transmite un parámetro (PC) idéntico.

6. Procedimiento de acuerdo con la reivindicación 5, en el que el aparato de prueba (300), para el cálculo del valor hash de comprobación (H'1), calcula en función del identificador (Z) un estado interno (IZ) en dependencia de la función hash, del parámetro (PC) y del bloque de datos de programa (PD1) y memoriza el estado interno (IZ).

7. Procedimiento de acuerdo con una de las reivindicaciones 1 a 4, en el que el aparato de prueba (300) transmite un parámetro único (PS1) al aparato (100) y otro parámetro único (PS2) a al menos otro aparato (102) y se transmiten el parámetro único (PS1) enviado y el otro parámetro único (PS2) enviado dentro de una ventana de tiempo que puede predeterminarse.

8. Procedimiento de acuerdo con una de las reivindicaciones 1 a 4, en el que el aparato de pruebas (300) transmite un parámetro único (PS1) al aparato (100) y otro parámetro único (PS2) a al menos otro aparato (102), en función del identificador (Z)

9. Procedimiento de acuerdo con una de las reivindicaciones precedentes, en el que la integridad del aparato (100) se confirma mediante el aparato de prueba (300) cuando el cálculo y transmisión del valor hash (H1) se realiza mediante el aparato (100) dentro de una ventana de tiempo de respuesta que puede predeterminarse de forma individual.

10. Procedimiento de acuerdo con una de las reivindicaciones 7 u 8, en el que el aparato de pruebas (300) calcula el valor hash de comprobación (H'1) antes o después de la transmisión del parámetro único (PS1).

11. Sistema de prueba para comprobar una integridad de datos de propiedades (ED1) de un aparato (100), incluyendo los datos de propiedades (ED1) un bloque de datos de programa (PD1) y un bloque de datos de configuración (KD1), mediante un aparato de pruebas (300) dentro de una red (400), que incluye el aparato (100), el aparato de pruebas (300) y al menos otro aparato (102) con otros datos de propiedades (ED2), incluyendo los otros datos de propiedades (ED2) otro bloque de datos de programa (PD2) y otro bloque de datos de configuración (KD2), incluyendo:
    - una estación (200) para proporcionar los datos de propiedades (ED1) y los otros datos de propiedades (ED2), estando acoplada la estación (200) al menos con el aparato de prueba (300) mediante la red (400);
    - el aparato de prueba (300) para evaluar los datos de propiedades (ED1) y los otros datos de propiedades (ED2) en cuanto a al menos una identidad del bloque de datos de programa (PD1) y del otro bloque de datos de programa(PD2), y
    asignación de un identificador (Z) en función de un resultado de la evaluación, determinación de un parámetro (P, PC, PS1), que es adecuado para influir sobre un valor hash (H1) y un valor hash de comprobación (H'1) de una función hash, configurándose el parámetro (P, PC, PS1) en función del identificador (Z),
    transmisión del parámetro (P, PC, PS1) al aparato (100),
    cálculo del valor hash de comprobación (H'1) en dependencia de la función hash y de una cadena de caracteres, incluyendo la cadena de caracteres el parámetro (P, PC, PS1) y los datos de propiedades (ED1) y pudiendo predeterminarse una secuencia dentro de la cadena de caracteres y confirmación de la integridad de los datos de propiedades (ED1), en el caso de que el valor hash (H1) coincida con el valor hash de comprobación (H'1);
    - el aparato (100) para
    calcular el valor hash (H1) en dependencia de la función hash y de una cadena de caracteres, incluyendo la cadena de caracteres el parámetro (P, PC, PS1) y los datos de propiedades (ED1) y pudiendo predeterminarse una secuencia dentro de la cadena de caracteres y
    transmisión del valor hash (H1) al aparato de prueba (300).

12. Sistema de prueba de acuerdo con la reivindicación 11,
además al menos otra unidad adicional para utilizarla en una de las etapas del procedimiento de acuerdo con las reivindicaciones 2 a 10.

13. Aparato de pruebas de acuerdo con la reivindicación 11 ó 12,
en el que el aparato de pruebas (300) y la estación (200) están configurados como unidad común.

FIG 1

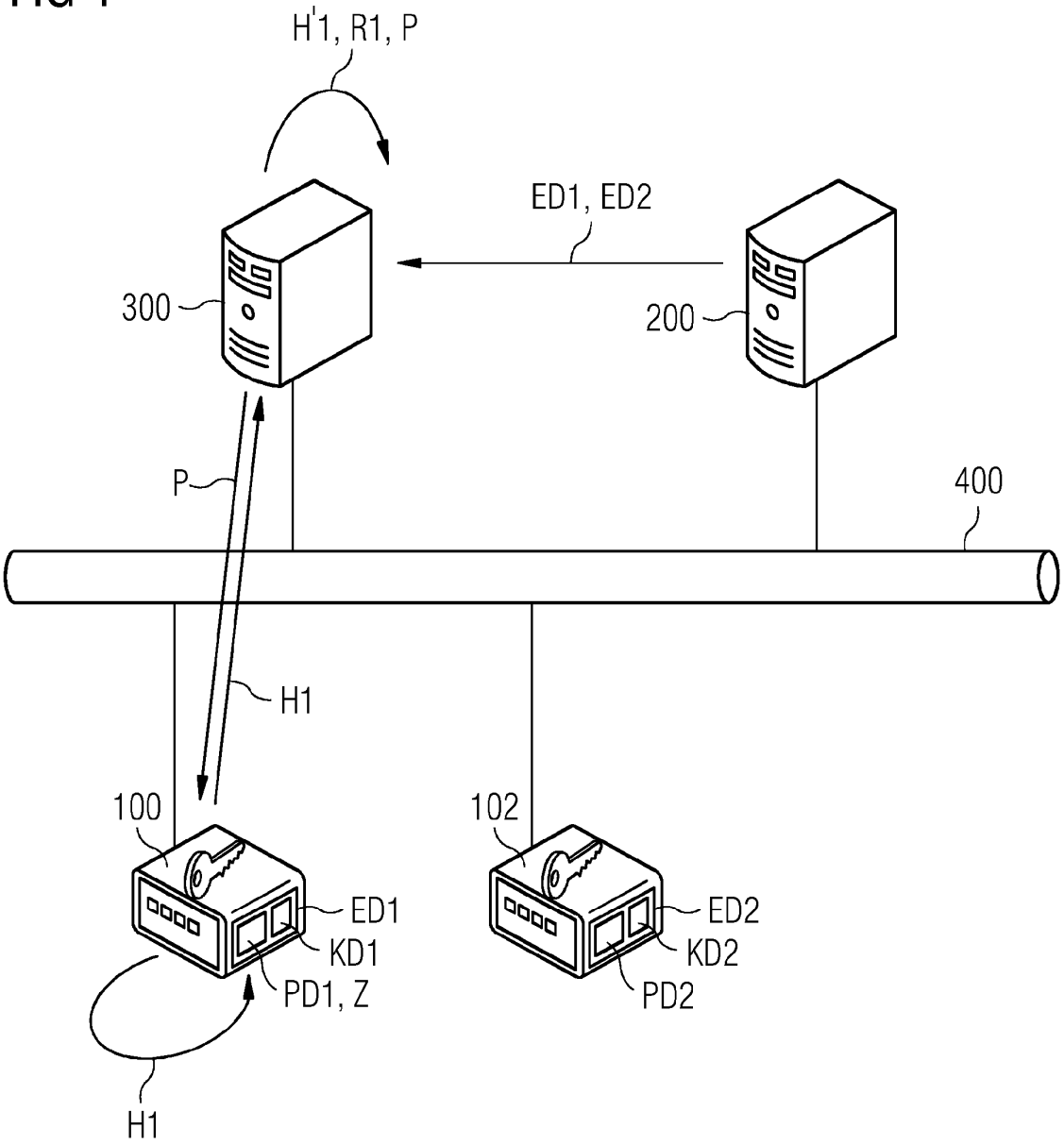

H'1, R1, P
ED1, ED2
300
200
P
400
H1
100
102
ED1
KD1
PD1, Z
ED2
KD2
PD2
H1

FIG 2

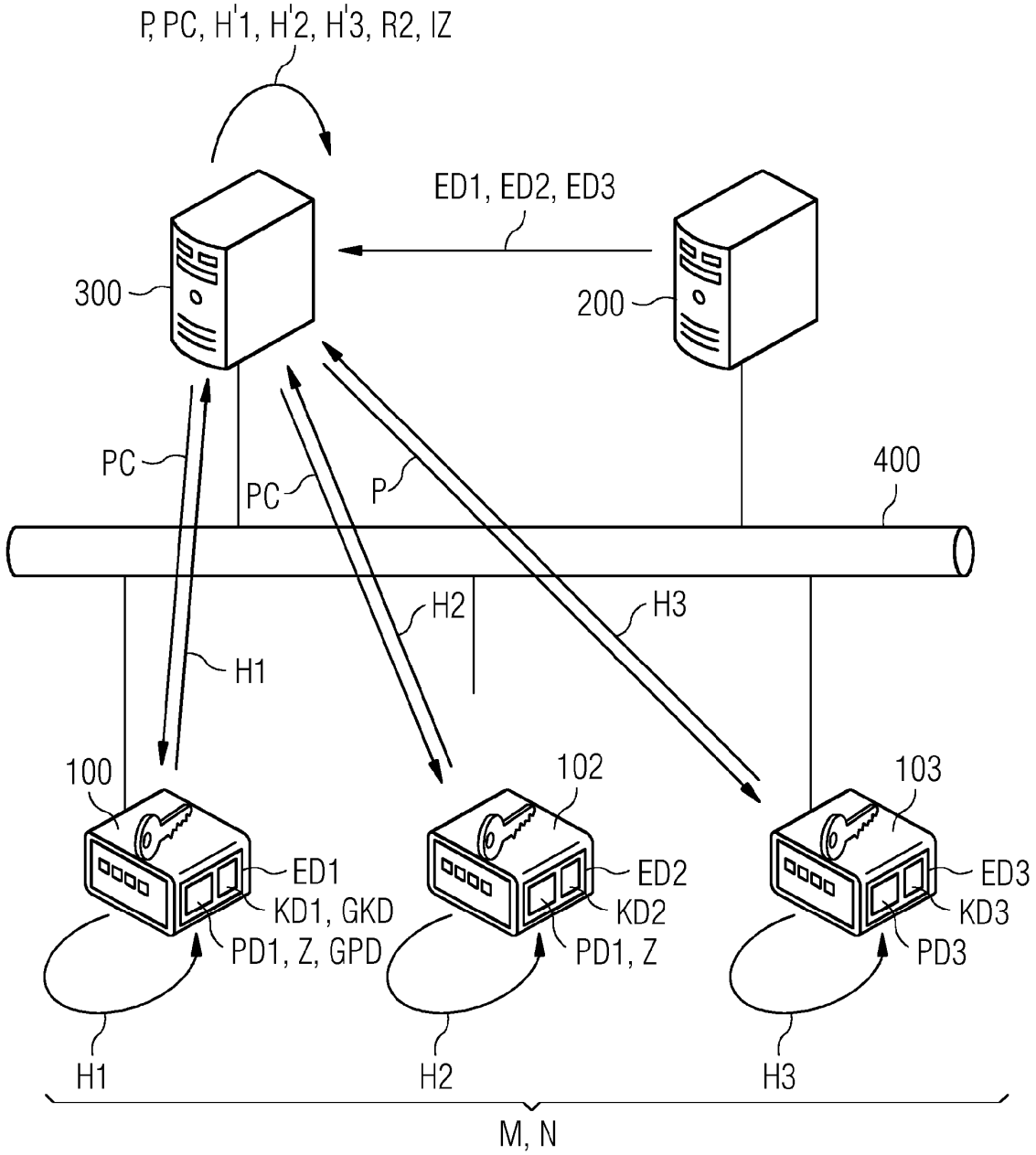
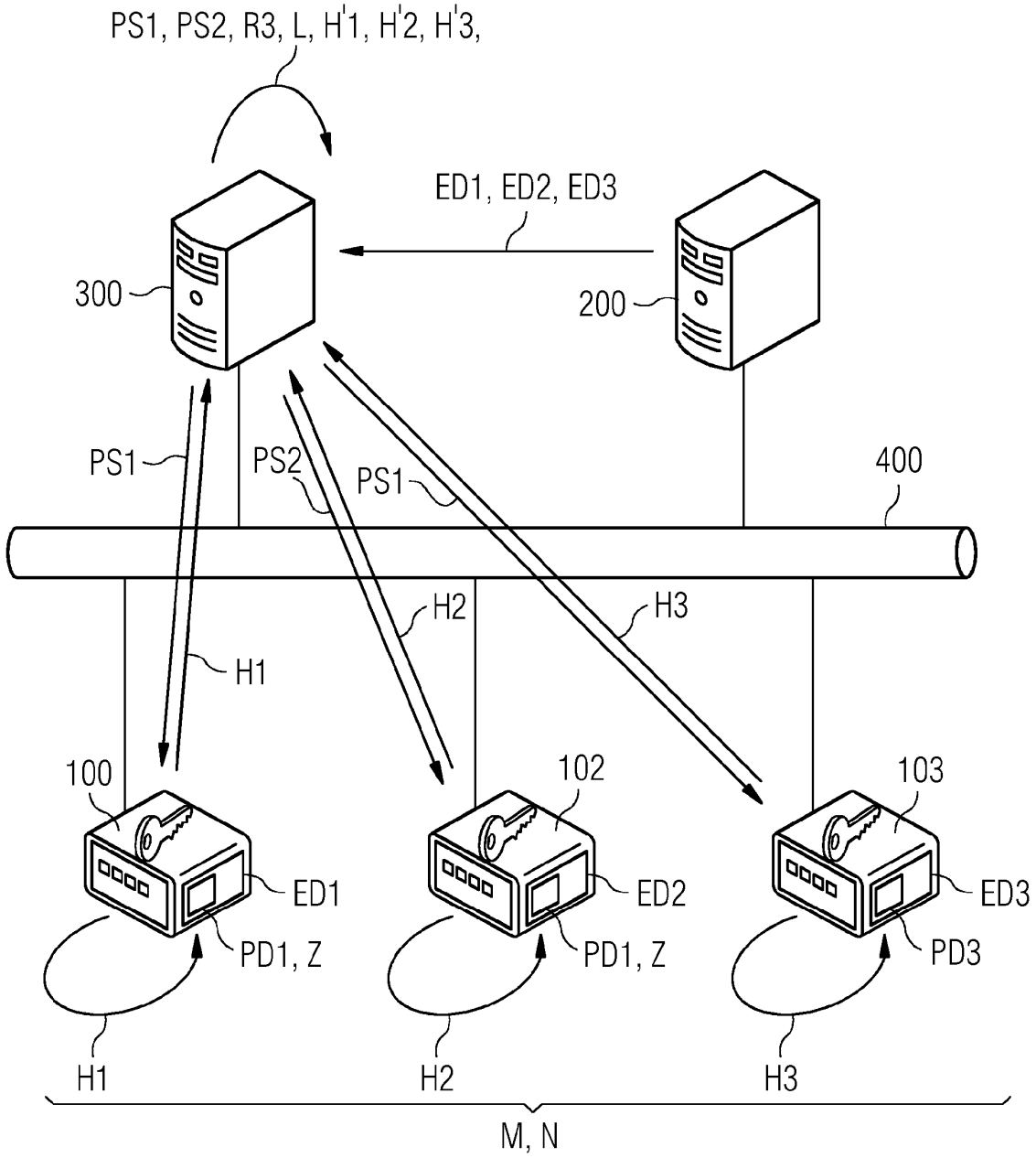

P, PC, H'1, H'2, H'3, R2, IZ
ED1, ED2, ED3
300
200
PC
PC
P
400
H2
H3
H1
100
102
103
ED1
KD1, GKD
PD1, Z, GPD
ED2
KD2
PD1, Z
ED3
KD3
PD3
H1
H2
H3
M, N

FIG 3


PS1, PS2, R3, L, H'1, H'2, H'3,
ED1, ED2, ED3
300
200
PS1
PS2
PS1
400
H2
H3
H1
100
102
103
ED1
ED2
ED3
PD1, Z
PD1, Z
PD3
H1
H2
H3
M, N