(19) Deutsches Patent- und Markenamt

(10) **DE 10 2016 110 414 B4** 2025.05.22

(12) # Patentschrift

(21) Aktenzeichen: **10 2016 110 414.0**
(22) Anmeldetag: **06.06.2016**
(43) Offenlegungstag: **08.12.2016**
(45) Veröffentlichungstag der Patenterteilung: **22.05.2025**

(51) Int Cl.: ***H04L 43/00*** (2022.01)
H04L 9/32 (2006.01)
G05B 19/418 (2006.01)
G05B 23/02 (2006.01)
G06F 21/50 (2013.01)
G06F 21/57 (2013.01)
H04L 67/12 (2022.01)

Innerhalb von neun Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 1 Patentkostengesetz).

(30) Unionspriorität:
14/732,228 **05.06.2015** **US**

(73) Patentinhaber:
Fisher-Rosemount Systems, Inc., Round Rock, Tex., US

(74) Vertreter:
Meissner Bolte Patentanwälte Rechtsanwälte Partnerschaft mbB, 80538 München, DE

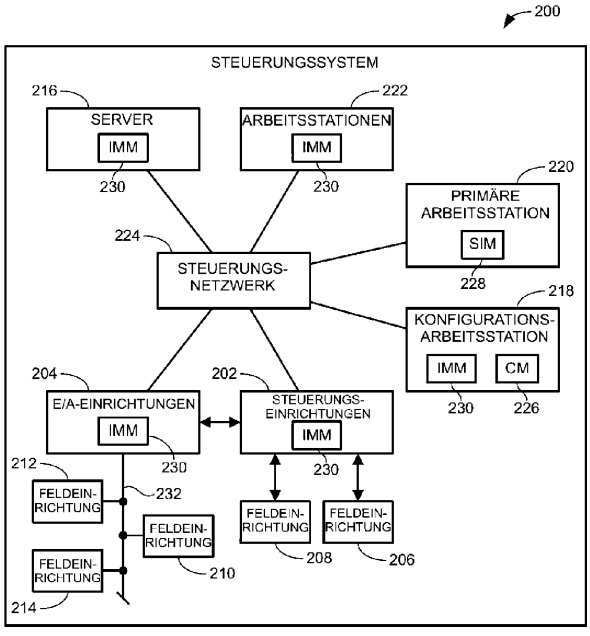
(72) Erfinder:
Mixer, Robert A., Cedar Park, Tex., US

(56) Ermittelter Stand der Technik:

DE	**698 32 096**	**T2**
US	**2012 / 0 151 206**	**A1**
US	**2013 / 0 150 985**	**A1**

(54) Bezeichnung: **VERFAHREN UND VORRICHTUNGEN ZUR STEUERUNG DER KOMMUNIKATION VON ENDPUNKTEN IN EINEM INDUSTRIEUNTERNEHMENSSYSTEM AUF DER BASIS VON INTEGRITÄT**

(57) Hauptanspruch: Vorrichtung, umfassend:
einen Integritätsmessungskomparator zum Vergleichen einer Integritätsmessung mit einem Referenzwert, wobei die Integritätsmessung durch einen ersten Endpunkt in einem ersten Netzwerk eines Industrieunternehmenssystems basierend auf einem Status des ersten Endpunkts erzeugt wird, wobei der Referenzwert einem vertrauenswürdigen Status des Endpunkts entspricht, wobei der Zustand des Endpunkts mindestens einem von Software oder Firmware entspricht, die auf den Endpunkt geladen ist, Konfigurationsdaten, die dem Endpunkt zugewiesen sind, oder einer Hardware oder Peripherie, die mit dem Endpunkt gekoppelt ist, entspricht, und wobei der erste Endpunkt einer Gruppe von zweiten Endpunkten in einem Teilnetz des ersten Netzes entspricht, wobei die Integritätsmessung einer Kombination von verschiedenen Integritätsmessungen entspricht, die der Gruppe von zweiten Endpunkten in dem Teilnetz entsprechen; und
eine Autorisierungssteuerungseinrichtung zum Ermöglichen des Kommunikationszugriffs für den Endpunkt auf das Netzwerk basierend auf dem Vergleich der Integritätsmessung mit dem Referenzwert.



Beschreibung

GEBIET DER OFFENBARUNG

[0001] Diese Offenbarung bezieht sich im Allgemeinen auf industrielle Steuerung und insbesondere auf Verfahren und Vorrichtungen zur Steuerung der Kommunikation von Endpunkten in einem Industrieunternehmenssystem auf der Basis von Integrität.

HINTERGRUND

[0002] Prozesssteuerungssysteme wie solche, die in Unternehmen der Chemie-, Öl- oder anderen Industriezweigen verwendet werden, enthalten üblicherweise eine oder mehrere Prozesssteuerungseinrichtungen, die über analoge-, digitale oder kombiniert analog/digitale Busse kommunikativ mit einer oder mehreren Feldeinrichtungen verbunden sind. Die Feldeinrichtungen, die zum Beispiel Ventile, Ventilstellungsregler, Schalter und Sender (z. B. Temperatur-, Druck- und Flussratensensoren) sein können, führen Prozesssteuerungsfunktionen innerhalb des Prozesses aus, zum Beispiel das Öffnen oder Schließen von Ventilen und das Messen von Prozesssteuerungsparametern. Die Prozesssteuerungseinrichtungen empfangen Signale, die aussagekräftig in Bezug auf Prozessmessungen sind, welche durch die Feldeinrichtungen erfolgt sind, und verarbeiten dann diese Informationen und/oder kommunizieren die Informationen zu anderen Steuerungseinrichtungen, Arbeitsstationen oder anderen Komponenten innerhalb des Steuerungssystems. Des Weiteren können mehrere Steuerungssysteme Subsysteme eines auf höherer Ebene angesiedelten Unternehmenssystems sein, in dem Informationen von den separaten Subsystemen untereinander kommuniziert und/oder durch ein Beaufsichtigungssystem überwacht werden.

[0003] Relevanten Stand der Technik bilden die Druckschriften US 2012/0151206 A1 und DE 698 32 096 T2, die sich jeweils mit der Überprüfung der Integrität eines Netzwerk-Knotens befassen. Hierbei gibt die US 2012/0151206 A1 insbesondere Hinweise auf die Durchführung eines Soll-Ist-Vergleiches des Status eines Endpunktes bzw. Knotens und auf eine davon abhängige Freigabe einer Kommunikation. Die DE 698 32 096 T2 gibt Hinweise auf eine Zurückstufung eines Knotens im Ergebnis der Erkennung dessen, dass ein Fehler im Knoten einen Schwellenwert überschreitet.

ZUSAMMENFASSUNG

[0004] Verfahren und Vorrichtungen zur Steuerung der Kommunikation von Endpunkten in einem auf Integrität basierenden Industrieunternehmenssystem werden offenbart. Eine beispielhafte Vorrichtung enthält einen Integritätsmessungskomparator zum Vergleichen einer Integritätsmessung mit einem Referenzwert. Die Integritätsmessung wird durch einen Endpunkt in einem Netzwerk eines Industrieunternehmenssystems basierend auf einem Status des Endpunkts erzeugt. Der Referenzwert entspricht einem vertrauenswürdigen Status des Endpunkts. Die beispielhafte Vorrichtung enthält auch eine Autorisierungssteuerungseinrichtung zum Ermöglichen des Kommunikationszugriffs für den Endpunkt auf das Netzwerk basierend auf dem Vergleich der Integritätsmessung mit dem Referenzwert.

[0005] Ein beispielhaftes Verfahren umfasst das Empfangen einer Integritätsmessung von einem Endpunkt in einem Netzwerk eines Industrieunternehmenssystems. Die Integritätsmessung wird durch den Endpunkt basierend auf einem Status des Endpunkts erzeugt. Das beispielhafte Verfahren umfasst des Weiteren das Vergleichen der Integritätsmessung mit einem Referenzwert, der einem vertrauenswürdigen Status des Endpunkts entspricht. Das beispielhafte Verfahren umfasst auch das Ermöglichen des Kommunikationszugriffs für den Endpunkt auf das Netzwerk basierend auf dem Vergleich.

[0006] Ein beispielhafter Herstellungsartikel enthält Anweisungen, die bei ihrer Ausführung bewirken, dass eine Maschine mindestens eine Integritätsmessung von einem Endpunkt in einem Netzwerk eines Industrieunternehmenssystems empfängt. Die Integritätsmessung wird durch den Endpunkt basierend auf einem Status des Endpunkts erzeugt. Die Anweisungen bewirken des Weiteren, dass die Maschine die Integritätsmessung mit einem Referenzwert vergleicht, der einem vertrauenswürdigen Status des Endpunkts entspricht. Die Anweisungen bewirken auch, dass die Maschine den Kommunikationszugriff für den Endpunkt auf das Netzwerk basierend auf dem Vergleich ermöglicht.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

Fig. 1 ist ein Blockdiagramm einer beispielhaften Hierarchie von Komponenten in einem Industrieunternehmenssystem.

Fig. 2 ist ein Blockdiagramm eines beispielhaften Steuerungssystems entsprechend der Steuerungssystemebene der beispielhaften Hierarchie aus **Fig.** 1.

Fig. 3 ist ein Blockdiagramm eines beispielhaften Beaufsichtigungssystems entsprechend der Beaufsichtigungssystemebene der beispielhaften Hierarchie aus **Fig.** 1.

Fig. 4 stellt eine beispielhafte Implementierungsweise des beispielhaften Konfigurationsmoduls aus **Fig.** 2 dar.

Fig. 5 stellt eine beispielhafte Implementierungsweise der beispielhaften Integritätsmessungsmodule aus **Fig.** 2 und/oder 3 dar.

Fig. 6 stellt eine beispielhafte Implementierungsweise der beispielhaften Systemintegritätsüberwachungsmodule aus **Fig.** 2 und/oder 3 dar.

Fig. 7 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens, das zur Implementierung des beispielhaften Konfigurationsmoduls aus **Fig.** 2 und/oder 4 durchgeführt werden kann.

Fig. 8-10 sind Ablaufdiagramme zur Darstellung von beispielhaften Verfahren, die zur Implementierung der beispielhaften Integritätsmessungsmodule aus **Fig.** 2, 3 und/oder 5 durchgeführt werden können.

Fig. 11-16 sind Ablaufdiagramme zur Darstellung von beispielhaften Verfahren, die zur Implementierung der beispielhaften Systemintegritätsüberwachungsmodule aus **Fig.** 2, 3 und/oder 6 durchgeführt werden können.

Fig. 17 ist eine schematische Darstellung einer beispielhaften Prozessorplattform, die verwendet und/oder programmiert werden kann, um das beispielhafte Verfahren aus **Fig.** 7 auszuführen, und/oder, noch allgemeiner, um das beispielhafte Konfigurationsmodul aus **Fig.** 2 und/oder 4 zu implementieren.

Fig. 18 ist eine schematische Darstellung einer beispielhaften Prozessorplattform, die verwendet und/oder programmiert werden kann, um die beispielhaften Verfahren aus **Fig.** 8-10 auszuführen, und/oder, noch allgemeiner, um die beispielhaften Integritätsmessungsmodule aus **Fig.** 2, 3 und/oder 5 zu implementieren.

Fig. 19 ist eine schematische Darstellung einer beispielhaften Prozessorplattform, die verwendet und/oder programmiert werden kann, um die beispielhaften Verfahren aus **Fig.** 11-16 auszuführen, und/oder, noch allgemeiner, um die beispielhaften Systemintegritätsüberwachungsmodule aus **Fig.** 2, 3 und/oder 6 zu implementieren.

DETAILLIERTE BESCHREIBUNG

[0007] Wenn hier die Rede von einem Industrieunternehmenssystem ist, so bezieht sich das auf die Rohstoffe und die zugeordneten physischen Prozesse, auf die Feldeinrichtungen, die mit solchen Prozessen interagieren, auf entsprechende Steuerungssysteme, die solche Prozesse steuern, und auf auf höherer Ebene angesiedelte administrative Systeme eines Industrieunternehmens. Damit ist ein Industrieunternehmenssystem ein System von hierarchisch miteinander in Beziehung stehenden Systemen (z. B. gibt es mehrere Ebenen), die üblicherweise über entsprechende Netzwerke miteinander verbunden sind. Wenn hier die Rede von einem Industriesystem-Computerendpunkt (oder Endpunkt) ist, bezieht sich das auf jede Computerkomponente, die als ein Knoten in einem Netzwerk (auf jeder Ebene) innerhalb eines Industrieunternehmenssystems dient, um mit anderen solchen Komponenten zu kommunizieren. Damit umfasst hier ein Endpunkt individuelle Computereinrichtungen (z. B. Feldeinrichtungen, Steuerungseinrichtungen, E/A-Einrichtungen, in sich geschlossene Plug-In-Karten für Steuerungseinrichtungen und/oder E/A-Einrichtungen, Arbeitsstationen, Server usw.) sowie Systeme oder Gruppen solcher Einrichtungen, die als ein integriertes gesamtes untergeordnetes System eines auf höherer Ebene angesiedelten Systems betrieben werden (z. B. ein modulares Steuerungssystem innerhalb eines größeren Steuerungssystems, ein Steuerungssystem innerhalb eines Beaufsichtigungssystems usw.). Üblicherweise bauen die Endpunkte in einem Industerieunternehmenssystem die Kommunikation untereinander auf, basierend entweder auf ihren Konfigurationsinformationen (bei leistungsfähigeren Endpunkten) oder auf ihrer physischen Konnektivität (bei weniger leistungsfähigen Endpunkten). Im Kontext von Computer- und Datensicherheit basiert der Aufbau der Kommunikation zwischen solchen Endpunkten auf einer Form des impliziten Vertrauens, weil kein Endpunkt wirklich weiß, ob die Kommunikation von anderen Endpunkten sicher und zuverlässig ist oder durch eine Art von Sicherheitsverstoß oder -fehler kompromittiert wurde. Insbesondere hängt der Aufbau der Kommunikation zwischen den Endpunkten zum Betreiben eines Industriebetriebs davon ab, dass jeder Endpunkt darauf vertraut, dass der andere Endpunkt (a) legitimiert ist (d. h. dass der andere Endpunkt wirklich dem entspricht, was er vorgeblich ist) und (b) mit voller Integrität arbeitet (d. h. dass der andere Endpunkt ordnungsgemäß funktioniert und dass die gesendete Kommunikation sowie die für diese Kommunikation erzeugten Daten nicht kompromittiert wurden).

[0008] Bei einigen bekannten Industriesystemendpunkten basiert das Vertrauen in die Sicherheit und/oder Zuverlässigkeit der Kommunikation auf identifizierenden Informationen (z. B. Seriennummern, Hardware-Adressen oder IP-Adressen), die jedem Endpunkt zugeordnet sind, und auf der Kenntnis des Kommunikationsprotokolls, das zur Kommunikation verwendet wird. Während dies der Bestätigung der Legitimität eines konkreten Endpunkts dient (z. B. ob ein Endpunkt wirklich dem entspricht, was er vorgeblich ist), sind derartige Ansätze unzureichend, um eine zuverlässige Auskunft über die Integrität von jedem Endpunkt zu geben (z. B. ob ein Endpunkt erwartungsgemäß funktioniert oder ob er kompromittiert wurde, wodurch die Zuverlässigkeit seines Betriebs in Frage gestellt würde). In der Ver-

gangenheit wurde das auf Integrität basierende Vertrauen größtenteils auf der Grundlage der Zusicherung der Produktqualität durch die Hersteller der an der Kommunikation beteiligten Computerendpunkte angenommen. Das heißt, die Hersteller folgen üblicherweise Standards (z. B. International Organization for Standardization (ISO) 9000), die Qualitätssicherungsprozesse für die Entwicklung und Erprobung von Software, Firmware und/oder Hardware definieren. Im Ergebnis dessen gibt es ein gewisses Maß an Vertrauen, dass die Einrichtungen (z. B. Endpunkte), die von solchen Herstellern produziert werden, entsprechend ihrer Konstruktionsvorgaben funktionieren (d. h. mit vertrauenswürdiger Integrität).

[0009] Während die Qualitätszusicherungen der Hersteller ein bestimmtes Maß an Vertrauen in die Integrität der von den Herstellern produzierten Computereinrichtungen schafft, verlieren diese Zusicherungen im Laufe der Zeit an Gewicht, da die Möglichkeit eines Sicherheitsverstoßes oder anderer Fehlerursachen besteht, nachdem die Einrichtung den Einflussbereich des Herstellers verlassen hat. So besteht beispielsweise die Möglichkeit, dass eine Computereinrichtung zwischen dem Zeitpunkt der Auslieferung durch einen Hersteller und der Zustellung zu einem Endbenutzer manipuliert wird. Des Weiteren besteht selbst nach der Entgegennahme einer Computereinrichtung durch einen Endbenutzer die Möglichkeit, dass die Einrichtung gehackt und durch schädlichen Code verändert wird. Obwohl üblicherweise Sicherheitsmaßnahmen ergriffen werden, um derartige Angriffe zu verringern und/oder zu erkennen, kann die schädliche Software den Betrieb der Computereinrichtung beeinträchtigen, wenn ein bestimmter Angriff nicht erkannt wird. Demzufolge kann das Vertrauen in die Integrität einer derartigen Einrichtung und/oder der von dieser Einrichtung gesendeten Kommunikation fehl am Platze sein. Dementsprechend besteht eine Notwendigkeit zur Verifizierung der Integrität von Endpunkten, wenn diese neu erworben, konfiguriert und in Betrieb genommen werden, sowie zur Überwachung der Integrität von Endpunkten in einem Industrieunternehmenssystem im Laufe der Zeit, um alle potenziellen Bedrohungen für die oder Defekte in der Integrität solcher Endpunkte zu erkennen und geeignete Reaktionsmaßnahmen basierend auf jeder entsprechenden Änderung in dem Maß an Vertrauen zu implementieren, das der Kommunikation von solchen Endpunkten entgegengebracht wird.

[0010] Die hier offenbarten Beispiele überwinden die oben identifizierten Probleme, indem die Endpunkte in einem Netzwerk gezwungen werden, berechnete oder gemessene Werte bereitzustellen, die aussagekräftig in Bezug auf bestimmte Aspekte ihres aktuellen Status sind, um diese mit bekannten guten Werten zu vergleichen, welche dem Status der Endpunkte in einem vertrauenswürdigen Status (d. h. einem Status der Integrität) entsprechen, bevor den Endpunkten die Kommunikation in dem entsprechenden Netzwerk ermöglicht wird. Die durch einen Endpunkt in Bezug auf seinen eigenen Status erzeugten Messungen werden hier als Integritätsmessungen bezeichnet. Die Integritätsmessungen werden, in einigen Beispielen, erzeugt, indem eine oder mehrere Prüfsummen zu unterschiedlichen Aspekten des Endpunkts berechnet werden, die aussagekräftig in Bezug auf den Betrieb, die Sicherheit und/oder die Zuverlässigkeit des Endpunkts sein können. Damit werden, in einigen Beispielen, die Integritätsmessungen hier als berechnet beschrieben. Das heißt, dass Daten, die unterschiedlichen Aspekten des Endpunkts zugeordnet sind, durch einen kryptografischen Prüfsummenalgorithmus geschickt werden, um einen Wert zu erzeugen, der als eine Integritätsmessung oder zur Erstellung einer Integritätsmessung (zusammen mit den Prüfsummen, die für andere Aspekte des Endpunkts berechnet wurden) verwendet werden kann. In einigen Beispielen erfolgt die Erzeugung der Integritätsmessungen basierend auf dem Software-Stack des Endpunkts. In einigen Beispielen erfolgt die Erzeugung der Integritätsmessungen basierend auf der Konfiguration des Endpunkts innerhalb eines Steuerungssystems. In einigen Beispielen erfolgt die Erzeugung der Integritätsmessungen basierend auf den Peripherieeinrichtungen, die dem Endpunkt zugeordnet sind. In einigen Beispielen erfolgt die Erzeugung der Integritätsmessungen durch einen eigenständigen Sicherheits-Chip, der dem Endpunkt zugeordnet ist. Darüber hinaus erfolgt, in einigen Beispielen, die Erzeugung der Integritätsmessungen während der Boot-Zeit des Endpunkts. In anderen Beispielen erfolgt die Erzeugung der Integritätsmessungen während der Laufzeit des Endpunkts. In einigen Beispielen, in denen der Endpunkt einem Subsystem von mehreren Computereinrichtungen (z. B. Endpunkten) entspricht, sind die Integritätsmessungen Kombinationen aus mehreren auf niedrigerer Ebene angesiedelten Integritätsmessungen, die jeder individuellen Einrichtung innerhalb des Subsystems zugeordnet sind.

[0011] Jede durch einen Endpunkt erzeugte Integritätsmessung kann mit entsprechenden Referenzwerten verglichen werden, von denen bekannt ist, dass sie einem vertrauenswürdigen Status des Endpunkts entsprechen. In einigen Beispielen basieren die Referenzwerte auf Integritätsmessungen, die erzeugt wurden, als ein Endpunkt erstmalig erworben, installiert und/oder konfiguriert wurde. In einigen Beispielen erfolgt die Definition der Referenzwerte basierend auf Werten, die durch den Hersteller des Endpunkts bereitgestellt wurden. Wenn ein Vergleich der Integritätsmessungen mit den Referenzwerten eine Übereinstimmung anzeigt, wird der nachgeordnete Endpunkt als in einem vertrauenswürdigen Sta-

tus befindlich bestätigt. Das heißt, dass die Software/Firmware, Hardware/Peripherieeinrichtungen, Konfiguration und/oder andere gemessene Aspekte des Endpunkts erwartungsgemäß sind, sodass die Kommunikation von dem Endpunkt vertrauenswürdig und/oder zuverlässig ist. In solchen Situationen kann dem Endpunkt basierend auf dessen verifizierter Integrität die Zulassung zur vollen Kommunikation in einem entsprechenden Netzwerk gewährt werden. Wenn jedoch die Integritätsmessungen nicht mit den entsprechenden Referenzwerten übereinstimmen, gerät die Integrität des Endpunkts unter Verdacht, und eine geeignete Reaktionsmaßnahme kann implementiert werden. In einigen dieser Beispiele kann der Typ und/oder der Grad des Kommunikationszugriffs des Endpunkts eingeschränkt und etwas geringer als der vollständige Kommunikationszugriff definiert werden. In einigen Beispielen basiert die Festlegung des Grads oder Typs des einem Endpunkt bereitgestellten Kommunikationszugriffs auf durch den vorgeordneten Endpunkt bereitgestellten Autorisierungsinformationen, die verwendet werden, um geheime und/oder öffentliche Werte oder Schlüssel zu erzeugen oder zu definieren, die zur Vervollständigung der Kommunikation zwischen Endpunkten verwendet werden. Ein derartiger integritätsbasierter Verifizierungsprozess kann für neue Endpunkte implementiert werden, die zu einem Netzwerk hinzugefügt werden (um sicherzustellen, dass der Endpunkt legitim ist und erwartungsgemäß funktioniert, bevor er im Netzwerk registriert und/oder konfiguriert wird) und/oder kann auf Endpunkte angewendet werden, die bereits in Betrieb sind (um zu verifizieren, ob ein zuvor konfigurierter Endpunkt in einer Weise kompromittiert wurde, die seine Integrität in Frage stellt).

[0012] **Fig.** 1 ist ein Blockdiagramm einer beispielhaften Hierarchie 100 der Ebenen 102, 104, 106, 108, 110, 112 eines typischen Industrieunternehmenssystems. Die beispielhafte Hierarchie aus **Fig.** 1 folgt dem Purdue-Referenzmodell, das in ISA 95 (International Society of Automation 95) umrissen ist. Auf der untersten Ebene beginnt die Hierarchie 100 mit den physischen Prozessen 102 eines Herstellungs-/Verarbeitungsunternehmens. Beispielsweise entsprechen die physischen Prozesse 102 in einem Industrieunternehmenssystem den verarbeiteten Materialien und/oder Produkten sowie den Rohrleitungen, Tanks, Heizungen, Fördereinrichtungen und/oder anderen Ausrüstungsvermögenswerten, die direkt mit den verarbeiteten Materialien und/oder Produkten interagieren. Die nächsthöhere Ebene in der Hierarchie ist die Einrichtungsebene 104. Die Einrichtungsebene 104 entspricht intelligenten und nicht-intelligenten Feldeinrichtungen, die den physischen Prozess messen, überwachen und/oder beeinflussen. Beispielsweise umfasst die Einrichtungsebene 104 Ventile, Stellglieder, Temperatursensoren, Drucksensoren und so weiter.

[0013] Des Weiteren kann die Einrichtungsebene 104 Steuerungseinrichtungen und/oder E/A-Einrichtungen umfassen, die mit den Feldeinrichtungen interagieren und/oder diese steuern. Derartige Steuerungseinrichtungen und E/A-Einrichtungen dienen als Schnittstelle zwischen der Einrichtungsebene 104 und der Steuerungssystemebene 106, der nächsthöheren Ebene in der Hierarchie 100. Aufgabe der Steuerungssystemebene 106 ist die direkte Beaufsichtigung und Steuerung der Feldeinrichtungen und damit die Steuerung der physischen Prozesse. Ein Steuerungssystem auf der Steuerungssystemebene 106 kann ein verteiltes Steuerungssystem (Distributed Control System - DCS), ein Beaufsichtigungs- und Datenerfassungssystem (Supervisory and Data Acquisition - SCADA) und/oder ein anderes Prozesssteuerungssystem sein. Üblicherweise umfasst die Steuerungssystemebene 106 eine oder mehrere Steuerungseinrichtungen und E/A-Einrichtungen, die direkt mit Feldeinrichtungen verbunden sind, die der Einrichtungsebene 104 zugeordnet sind. Des Weiteren umfasst die Steuerungssystemebene 106 üblicherweise einen oder mehrere Server und/oder eine oder mehrere Arbeitsstationen, die eine Mensch-Maschine-Schnittstelle (Human Machine Interface - HMI) bereitstellen, mit der Bediener die Steuerung der physischen Prozesse konfigurieren, überwachen und/oder anpassen können. Jede(r) der Steuerungseinrichtungen, E/A-Einrichtungen, Arbeitsstationen und/oder Server in einem konkreten Steuerungssystem kann über ein Netzwerk des Steuerungssystems verbunden sein, das hier als das Steuerungsnetzwerk des Steuerungssystems bezeichnet wird.

[0014] Die Beaufsichtigungssystemebene 108 ist die nächsthöhere Ebene in der Hierarchie 100 und repräsentiert die Operationen, die der Beaufsichtigung der Herstellung/Verarbeitung des Unternehmens dienen. Somit umfasst die Beaufsichtigungssystemebene 108 üblicherweise ein System von Subsystemen (z. B. entsprechend einem oder mehreren Steuerungssystemen, die der Steuerungssystemebene 106 zugeordnet sind) und eine oder mehrere Arbeitsstationen und/oder einen oder mehrere Server zur Interaktion mit den Subsystemen. Ein solches System wird hier als ein Beaufsichtigungssystem bezeichnet. In einigen Beispielen kann ein Beaufsichtigungssystem die Herstellungs-/Verarbeitungsoperationen einer bestimmten Anlage innerhalb eines Unternehmens beaufsichtigen. Jede(r) der Arbeitsstationen, Server und anderen Subsysteme (z. B. andere Steuerungssysteme, die der Steuerungssystemebene 106 zugeordnet sind) kann über ein Netzwerk mit dem Beaufsichtigungssystem verbunden sein, das hier als ein Anlagennetzwerk oder Beaufsichtigungsnetzwerk des Beaufsichtigungssystems bezeichnet wird.

[0015] Oberhalb der Beaufsichtigungssystemebene 108 befindet sich die Geschäftssystemebene 110, die den geschäftsbezogenen Aktivitäten und Entscheidungen entspricht, die alle Aspekte des Unternehmens bestimmen und überwachen. Die höchste Ebene in der Hierarchie 100 des dargestellten Beispiels entspricht dem Internet 112. Obwohl technisch gesehen das Internet keine Ebene innerhalb eines bestimmten Unternehmens ist, erfolgt die Implementierung von Unternehmen oft auf der Basis der Kommunikation, die über das Internet bewerkstelligt wird. Als solches ist das Internet 112 zu Zwecken der Erklärung in einer Diskussion der Computersicherheit und -integrität repräsentiert, da es sich auf Industrieunternehmenssysteme bezieht.

[0016] Über die Länge der beispielhaften Hierarchie 100 aus **Fig.** 1 erstreckt sich eine Integritätsskala 114, welche die Bedeutung der Integrität oder Vertrauenswürdigkeit von Computerendpunkten auf jeder Ebene in der Hierarche 100 zur Erreichung des beabsichtigten Unternehmenszwecks repräsentiert. Wie im dargestellten Beispiel gezeigt wird, sind auf den hierarchischen Ebenen, die sich näher am physischen Prozess befinden, höhere Grade der Integrität erforderlich, während die Integrität weniger entscheidend wird, je weiter entfernt vom physischen Prozess eine Computereinrichtung implementiert ist. Wenn zum Beispiel bei den Operationen der untersten Ebene in einem Unternehmen (den physischen Prozessen 102 entsprechend) nicht darauf vertraut werden kann, dass sie erwartungsgemäß ablaufen, kann es kein Vertrauen in die resultierenden Produkte geben, die verarbeitet und/oder hergestellt werden, was in erster Linie der Hauptzweck der Einrichtung des Unternehmens ist. Allerdings entsprechen die physischen Ausrüstungsvermögenswerte und Materialien, die im Rahmen der physischen Prozesse 102 verarbeitet werden, üblicherweise Rohrleitungen, Tanks, Fülltrichtern und dergleichen sowie den zugehörigen Rohmaterialien. Damit ist, obwohl die korrekte Operation der physischen Prozesse 102 für ein Industrieunternehmenssystem entscheidend ist, unter dem Gesichtspunkt der Computer- und Datenintegrität die Einrichtungsebene 104 die wichtigste Ebene. Die Einrichtungsebene 104 ist am wichtigsten (d. h. die Integrität der entsprechenden Einrichtungen (z. B. Feldeinrichtungen, Steuerungseinrichtungen, E/A-Einrichtungen usw.) ist am wichtigsten), weil sie den Computereinrichtungen oder Endpunkten entspricht, die sich am nächsten zu den physischen Prozessen 102 befinden und direkt mit diesen interagieren.

[0017] Die nächste entscheidendste Ebene in der Hierarchie 100 unter dem Gesichtspunkt der Integrität ist die Steuerungssystemebene 106, weil ein Steuerungssystem die Konfiguration und Implementierung der Steuerungseinrichtungen, E/A-Einrichtungen und Feldeinrichtungen auf der Einrichtungsebene 104 definiert. Auf der Beaufsichtigungssystemebene 108 getroffene Entscheidungen haben üblicherweise keine direkten Auswirkungen auf die physischen Prozesse, sodass diese Ebene hinsichtlich der Integrität weniger entscheidend ist. Der Grad an Integrität, der auf jeder nachfolgenden Ebene bei Entfernung von den physischen Prozessen 102 erforderlich ist, nimmt zunehmend ab, bis das Internet erreicht ist, wo die Integrität nicht wichtig ist oder zumindest nicht erwartet werden kann, weil das Internet 112 ein offenes Netzwerk ist, zu dem so gut wie jeder eine Verbindung herstellen kann und über das jeder kommunizieren kann. Zur Gewährleistung und Aufrechterhaltung der Integrität von Endpunkten, die in niedrigeren Ebenen in der Hierarchie 100 implementiert sind, während weiterhin die Kommunikation zwischen den Ebenen ermöglicht wird, werden Einschränkungen für die Kommunikation zwischen Endpunkten in den unteren Ebenen der Hierarchie 100 und den Endpunkten in höheren Ebenen festgelegt. Insbesondere können der Charakter oder Typ des Kommunikationszugriffs sowie das Ausmaß der Kommunikation (z. B. die Anzahl und Typen der anderen Endpunkte, mit denen ein erster Endpunkt kommuniziert) streng geregelt sein, um die Wahrscheinlichkeit zu verringern, dass ein bestimmter Endpunkt korrumpiert wird (z. B. die Integrität aufgrund eines Sicherheitsverstoßes verliert). Gleichzeitig wird die Kommunikation eingeschränkt und besonders kontrolliert, um die Möglichkeit zu reduzieren, dass ein potentiell korrumpierter Endpunkt in einem System andere Endpunkte beeinflusst.

[0018] Zusätzlich zur Kontrolle der Kommunikation zwischen den Endpunkten zum Schutz der Endpunkte und zur Verringerung der Wahrscheinlichkeit, dass Endpunkte korrumpiert werden, wird - wie im Folgenden umfassender beschrieben -, in einigen Beispielen, die Integrität oder Vertrauenswürdigkeit von Endpunkten zu verschiedenen Zeitpunkten getestet, um insbesondere alle Endpunkte zu identifizieren, die möglicherweise kompromittiert wurden, um proaktiv zu reagieren, indem die Kommunikation von solchen Endpunkten eingeschränkt und/oder vollständig abgelehnt wird. Kurz gesagt fordern, in einigen Beispielen, die Endpunkte innerhalb einer gegebenen Ebene in der beispielhaften Hierarchie 100 die Registrierung in einem und/oder die Zulassung für ein Netzwerk an, das in der Hierarchie 100 eine Ebene höher angesiedelt ist. Um die Zulassung zu erhalten, damit der vollständige Kommunikationszugriff ermöglicht und/oder gestattet wird, muss, in einigen Beispielen, eine durch jeden Endpunkt bereitgestellte Messung, die aussagekräftig in Bezug auf dessen aktuellen Status ist, mit einem Referenzwert übereinstimmen, der dem Endpunkt entspricht, wenn sich dieser in einem vertrauenswürdigen Status befindet (z. B. einem Status der Integrität). Wenn die Messungen nicht mit dem entsprech-

enden Referenzwert übereinstimmen, kann die Zulassung zur Kommunikation vollständig abgelehnt werden, oder der Endpunkt kann in einen Abhilfemodus versetzt werden, in welchem der Charakter und/oder Umfang der Kommunikation signifikant eingeschränkt ist (z. B. begrenzt auf Software-/Firmware-Updates). In einigen hier offenbarten Beispielen werden integritätsbasierte Messungen implementiert, bevor die Autorisierung für die Kommunikation zwischen Endpunkten auf verschiedenen Ebenen in der Hierarchie 100 erfolgt. Insbesondere die Endpunkte auf den niedrigeren Hierarchieeben werden auf ihre Integrität getestet, bevor die Kommunikation mit den höheren Hierarchieebenen ermöglicht wird. Zum Beispiel muss, wie durch den Pfeil 116 in **Fig.** 1 dargestellt, eine Feldeinrichtung oder eine entsprechende Steuerungseinrichtung und/oder E/A-Einrichtung auf der Einrichtungsebene 104 möglicherweise eine Integritätsmeldung zu ihrem aktuellen Status bereitstellen, die mit einem entsprechenden Referenzwert übereinstimmt, bevor die Feldeinrichtung oder die entsprechende Steuerungseinrichtung und/oder E/A-Einrichtung auf einem Steuerungsnetzwerk eines Steuerungssystems auf der Steuerungssystemebene 106 kommunizieren kann. In gleicher Weise muss, wie durch den Pfeil 118 in **Fig.** 1 dargestellt, ein Steuerungssystem auf der Steuerungssystemebene 106 möglicherweise eine Integritätsmeldung zu seinem aktuellen Status bereitstellen, die mit einem entsprechenden Referenzwert übereinstimmt, bevor das Steuerungssystem auf einem Beaufsichtigungsnetzwerk eines Beaufsichtigungssystems auf der Beaufsichtigungssystemebene 108 kommunizieren kann. Dieser Verifizierungsprozess eines auf einer niedrigeren Ebene angesiedelten Endpunkts (wo die Integrität entscheidend ist) durch eine auf einer höheren Ebene angesiedelten Komponente (wo die Integrität weniger wichtig ist) dient der Sicherstellung, dass die Endpunkte, die sich am nächsten zu den physischen Prozessen 102 befinden, erwartungsgemäß funktionieren (d. h. mit Integrität).

[0019] **Fig.** 2 ist eine schematische Darstellung eines beispielhaften Steuerungssystems 200 entsprechend der Steuerungssystemebene 106 in der beispielhaften Hierarchie 100 aus **Fig.** 1. Das beispielhafte Steuerungssystem 200 aus **Fig.** 2 umfasst eine oder mehrere Prozesssteuerungseinrichtungen 202 sowie eine oder mehrere E/A-Einrichtungen 204, die als Schnittstelle zu den Feldeinrichtungen 206, 208, 210, 212, 214 entsprechend der Einrichtungsebene 104 der Hierarchie 100 aus **Fig.** 1 dienen. Außerdem umfasst das beispielhafte Steuerungssystem 200 aus **Fig.** 2 einen oder mehrere Server 216 sowie eine oder mehrere Bedienerstationen, Anwendungsstationen und/oder Arbeitsstationen (hier zusammengefasst als Arbeitsstationen bezeichnet). In dem dargestellten Beispiel ist eine Arbeitsstation innerhalb des Steuerungssystems 200 ausgewiesen oder dient als eine Konfigurationsarbeitsstation 218, während eine andere Arbeitsstation als eine primäre Arbeitsstation 220 dient (andere Arbeitsstationen sind durch die Bezugsziffer 222 repräsentiert). In dem dargestellten Beispiel sind die beispielhaften Steuerungseinrichtungen 202, die beispielhaften E/A-Einrichtungen 204, die beispielhaften Server 216 und die beispielhaften Arbeitsstationen 218, 220, 222 kommunikationsfähig über einen Kommunikationsbus und/oder ein lokales Netzwerk 224 gekoppelt, das hier im Allgemeinen als ein Steuerungsnetzwerk des Steuerungssystems 200 bezeichnet wird.

[0020] Die beispielhafte Konfigurationsarbeitsstation 218 enthält ein Konfigurationsmodul (Configuration Module - CM) 226 zum Erzeugen einer Konfigurationsdatei basierend auf Konfigurationsdaten, die durch einen Ingenieur, Bediener und/oder anderen Mitarbeiter bereitgestellt werden, die alle Parameter und die gesamte Logik definiert, die für die Implementierung der Einrichtungen (z. B. Endpunkte) innerhalb des Steuerungssystems 200 gewünscht sind bzw. ist. Sobald die Konfigurationsdatei erzeugt wurde, überträgt die Konfigurationsarbeitsstation 218 die Konfigurationsdatei zu jedem der Endpunkte des Steuerungsnetzwerks 224 in dem Steuerungssystem 200, wodurch jeder Endpunkt konfiguriert werden kann.

[0021] Die beispielhafte primäre Arbeitsstation 220 enthält ein Systemintegritätsüberwachungsmodul (System Integrity Monitor - SIM) 228 zur Überwachung, Autorisierung und/oder Steuerung des Kommunikationszugriffs von jedem Endpunkt im Steuerungsnetzwerk 224 des Steuerungssystems 200. Das heißt, dass - wie im Folgenden umfassender beschrieben - das SIM 228 der beispielhaften primären Arbeitsstation 220 aus **Fig.** 2 die Integrität der anderen Endpunkte innerhalb des Steuerungssystems 200 bestätigt und/oder verifiziert, um entweder den Kommunikationszugriff zu gestatten oder diesen Zugriff einzuschränken. Insbesondere empfängt das SIM 228, in einigen Beispielen, von jedem Endpunkt eine oder mehrere Integritätsmessungen, die durch ein Integritätsmessungsmodul (IMM) 230 erzeugt werden, das jedem Endpunkt zugeordnet ist. Wenn die Integritätsmessungen durch das SIM 228 empfangen wurden, vergleicht das SIM 228 die Integritätsmessungen mit einer Datenbank von Referenzwerten. Wenn die Messungen mit den Referenzwerten übereinstimmen, wird dem entsprechenden Endpunkt die vollständige Kommunikation mit anderen Endpunkten über das Steuerungsnetzwerk 224 gestattet. Wenn dagegen die Integritätsmessungen nicht mit den Referenzwerten übereinstimmen, kann das SIM 228 Abhilfemaßnahmen implementieren, um den Zugriff auf das Steuerungsnetzwerk 224 für die Kommunikation einzuschränken oder zu verweigern. In einigen

Beispielen überwacht das SIM 228 der primären Arbeitsstation 220 die Integrität der anderen Endpunkte des Steuerungsnetzwerks 224, um etwaige Änderungen zu erkennen und entsprechend zu reagieren. Obwohl die primäre Arbeitsstation 220 und die Konfigurationsarbeitsstation 218 als getrennte Arbeitsstationen dargestellt sind, kann, in einigen Beispielen, eine einzige Arbeitsstation beide Funktionen bedienen (z. B. sowohl das CM 226 als auch das SIM 228 implementieren).

[0022] Das beispielhafte Steuerungsnetzwerk 224 aus **Fig.** 2 kann unter Verwendung jedes gewünschten Kommunikationsmediums oder -protokolls implementiert werden. Zum Beispiel kann das beispielhafte Steuerungsnetzwerk 224 auf einem kabelgebundenen und/oder drahtlosen Ethernet-Kommunikationsschema basieren. Allerdings können alle anderen geeigneten Kommunikationsmedien und/oder Protokolle verwendet werden.

[0023] Die beispielhaften E/A-Einrichtungen 204 aus **Fig.** 2 sind über einen digitalen Datenbus 232 an eine Vielzahl von intelligenten Feldeinrichtungen 210, 212 214 gekoppelt. Die intelligenten Feldeinrichtungen 210, 212, 214 können Fieldbus-kompatible Ventile, Stellglieder, Sensoren usw. sein, wobei in diesem Fall die intelligenten Feldeinrichtungen 210, 212, 214 unter Verwendung des gut bekannten Foundation Fieldbus-Protokolls über den digitalen Datenbus 232 kommunizieren. Natürlich könnten auch andere Typen von intelligenten Feldeinrichtungen und Kommunikationsprotokollen verwendet werden. Zum Beispiel könnten die intelligenten Feldeinrichtungen 210, 212, 214 stattdessen Profibus- und/oder HART-kompatible Einrichtungen sein, die unter Verwendung der gut bekannten Profibus- und HART-Kommunikationsprotokokolle über den Datenbus 232 kommunizieren. Zusätzlich zu den beispielhaften intelligenten Feldeinrichtungen 210, 212, 214 können eine oder mehrere nicht-intelligente Feldeinrichtungen 206, 208 kommunikationsfähig mit der beispielhaften Steuerungseinrichtung 201 gekoppelt sein. Die nicht-intelligenten Feldeinrichtungen 206, 208 aus **Fig.** 2 können zum Beispiel herkömmliche 4-20-Milliampere- (mA) oder 0-10-Volt-Gleichstrom-(VDC)-Einrichtungen sein, die über jeweilige kabelgebundene Verbindungen mit der Steuerungseinrichtung 202 kommunizieren. Üblicherweise sind die Feldeinrichtungen 206, 208, 210, 212, 214 mit einer relativ beschränkten Verarbeitungsleistung konzipiert, die zur Implementierung der ihnen zugewiesenen Funktionen spezifiziert ist. Dementsprechend implementiert eine Feldeinrichtung möglicherweise nicht ein IMM 230, das eine Integritätsmessung erzeugt, die aussagekräftig in Bezug auf einen aktuellen Status der Feldeinrichtung ist. Allerdings können, in einigen Beispielen, eine oder mehrere Feldeinrichtungen 206, 208, 210, 212, 214 ein IMM 230 in gleicher Weise implementieren wie die anderen oben erwähnten Einrichtungen. In einigen solcher Beispiele kann die entsprechende Steuerungseinrichtung 202 oder E/A-Einrichtung 2024 ein getrenntes SIM implementieren, welches die Kommunikation zwischen den Feldeinrichtungen 206, 208, 210, 212, 214 und der jeweiligen Steuerungseinrichtung 202 oder E/A-Einrichtung 204 regelt. In anderen solchen Beispielen kann das durch die primäre Arbeitsstation 220 implementierte SIM 228 den Kommunikationszugriff solcher Feldeinrichtungen steuern.

[0024] Während **Fig.** 2 ein beispielhaftes Steuerungssystem 200 veranschaulicht, in dem die Verfahren und Vorrichtungen zur Steuerung der Kommunikationszulassungen basierend auf den nachstehend detailliert beschriebenen Integritätsmessungen vorteilhaft eingesetzt werden können, wird der durchschnittliche Fachmann auf dem Gebiet der Technik ohne weiteres anerkennen, dass der vorteilhafte Einsatz der hier offenbarten Lehren auf Wunsch auch in anderen Steuerungseinrichtungen mit höherer oder geringerer Komplexität (z. B. mit mehr als einer Steuerungseinrichtung, über mehr als einen geografischen Standort hinweg usw.) als im dargestellten Beispiel aus **Fig.** 2 möglich ist.

[0025] **Fig.** 3 ist eine schematische Darstellung eines beispielhaften Beaufsichtigungssystems 300 entsprechend der Beaufsichtigungssystemebene 108 in der Hierarchie 100 aus **Fig.** 1. Das beispielhafte Beaufsichtigungssystem 300 aus **Fig.** 3 umfasst das Steuerungssystem 200 aus **Fig.** 2 sowie ein weiteres Steuerungssystem 302 entsprechend der Steuerungssystemebene 106 der Hierarchie 100 aus **Fig.** 1. Außerdem umfasst das beispielhafte Beaufsichtigungssystem 300 aus **Fig.** 3 einen oder mehrere Server 304 und eine oder mehrere Arbeitsstationen 306, 308. In dem dargestellten Beispiel sind das Steuerungssystem 200, die Server 304 und die Arbeitsstationen 306, 308 kommunikationsfähig über einen Kommunikationsbus und/oder ein lokales Netzwerk 310 gekoppelt, das hier im Allgemeinen als ein Beaufsichtigungsnetzwerk für das Beaufsichtigungssystem 300 bezeichnet wird.

[0026] In dem dargestellten Beispiel arbeitet eine der Arbeitsstationen 306 als eine primären Arbeitsstation zur Überwachung und/oder Steuerung des Kommunikationszugriffs der anderen Endpunkte (z. B. der individuellen Server 304 und Arbeitsstationen 308 oder der Steuerungssysteme 200, 302 als Subsysteme des Beaufsichtigungssystems 300), die in gleicher oder ähnlicher Weise wie die primären Arbeitsstation 220 in dem Steuerungssystem 200 aus **Fig.** 2 mit dem Beaufsichtigungsnetzwerk 310 verbunden sind. Das heißt, in dem dargestellten Beispiel verfügt die primäre Arbeitsstation 306 über ein SIM 228 zum Empfangen von Integritätsmessungen, die aussagekräftig in Bezug auf den aktuellen Status

der anderen Endpunkte sind, und zum Vergleichen der Messungen mit Referenzwerten. In dem dargestellten Beispiel werden die den Servern 304 und/oder den Arbeitsstationen 308 zugeordneten Integritätsmessungen über entsprechende IMMs in jedem der Server 304 und/oder jeder der Arbeitsstationen 308 erzeugt. Obwohl jedes Steuerungssystem 200, 302 als ein Endpunkt behandelt werden kann (aus Sicht des Beaufsichtigungssystems 300), bewirkt die Tatsache, dass die Steuerungssysteme 200, 302 ein Subsystem aus mehreren Einrichtungen repräsentieren, dass kein einzelnes IMM die Integritätsmessungen erzeugen kann, die durch das SIM 228 der primären Arbeitsstation 306 analysiert werden soll. Dementsprechend erzeugt, in einigen Beispielen, das SIM 228 der primären Arbeitsstation 306 innerhalb jedes Steuerungssystems (z. B. die primären Arbeitsstation 220 des Steuerungssystems 200 aus **Fig.** 2) eine oder mehrere Integritätsmessungen, die aussagekräftig in Bezug auf einen Status des gesamten Subsystems sind und dem SIM 228 der primären Arbeitsstation 306 des Beaufsichtigungssystems 300 bereitgestellt werden. In einigen Beispielen basieren die Integritätsmessungen für ein gesamtes Steuerungssystem (als ein Endpunkt in einem Beaufsichtigungsnetzwerk) auf einer Kombination der Integritätsmessungen, die von jedem der IMMs 230 erhalten wurden, die an jedem Endpunkt des Steuerungsnetzwerks des Steuerungssystems implementiert sind.

[0027] Während **Fig.** 3 ein beispielhaftes Beaufsichtigungssystem 300 veranschaulicht, in dem die hier offenbarten Lehren vorteilhaft eingesetzt werden können, wird der durchschnittliche Fachmann auf dem Gebiet der Technik ohne weiteres anerkennen, dass der vorteilhafte Einsatz der hier offenbarten Lehren auf Wunsch auch in anderen Systemen mit höherer oder geringerer Komplexität (z. B. mit mehr Steuerungssystemen, über mehr als einen geografischen Standort hinweg usw.) als im dargestellten Beispiel aus **Fig.** 3 möglich ist.

[0028] **Fig.** 4 stellt eine beispielhafte Implementierungsweise des beispielhaften Konfigurationsmoduls (Configuration Module - CM) 226 aus **Fig.** 2 dar. In dem dargestellten Beispiel umfasst das CM 226 eine beispielhafte Benutzerschnittstelle 402, einen beispielhaften Konfigurationsdateigenerator 404, einen beispielhaften Referenzwertgenerator 406, eine beispielhafte Konfigurationsdatenbank 408 und eine beispielhafte Kommunikationsschnittstelle 410.

[0029] Das beispielhafte CM 226 aus **Fig.** 4 verfügt über die beispielhafte Benutzerschnittstelle 402, um Interaktionen zwischen dem CM 226 und einem Konfigurationsingenieur und/oder anderem Benutzer zu ermöglichen. Insbesondere kann ein Systemingenieur über die Benutzerschnittstelle 402 Parameter zuweisen, Steuerungslogik definieren, IP-Adressen angeben, Namen für physische Karten zuweisen und/oder alle anderen relevanten Konfigurationsdaten bereitstellen, die für die Konfiguration jedes Endpunkts innerhalb des Steuerungssystems erforderlich sind. Der beispielhafte Konfigurationsdateigenerator 404 des dargestellten Beispiels nimmt die eingegebenen Konfigurationsdaten und erzeugt eine Konfigurationsdatei für das Projekt oder Steuerungssystems, auf welches die Konfigurationsdaten angewendet werden sollen. In einigen Beispielen werden die Konfigurationsdaten und die resultierende Konfigurationsdatei in der beispielhaften Konfigurationsdatenbank 408 gespeichert.

[0030] In dem dargestellten Beispiel aus **Fig.** 4 verfügt das CM 226 über den beispielhaften Referenzwertgenerator 406 zur Berechnung von Konfigurationsreferenzwerten für jeden Endpunkt in dem Steuerungssystem 200, das basierend auf der Konfigurationsdatei auf Integrität überwacht werden soll (z. B. ein Integritätsmessungsmodul (IMM)). Das heißt, basierend auf dem dargestellten Beispiel aus **Fig.** 2, dass der beispielhafte Konfigurationsreferenzwertgenerator 406 des CM 266 Referenzwerte für jede der Steuerungseinrichtungen 202, der E/A-Einrichtungen 204, der Server 216 und der Arbeitsstationen 218, 220, 222 erzeugt. In einigen Beispielen erfolgt die Erzeugung oder Berechnung der Referenzwerte als eine kryptografische Prüfsumme der Konfigurationsdaten, die dem konkreten Endpunkt von Interesse zugeordnet sind. In solchen Beispielen entsprechen die berechneten Werte der Prüfsummen der ordnungsgemäßen Konfiguration jedes Endpunkts, weil die Berechnung der Prüfsummen basierend auf der originalen Konfigurationsdatei (durch einen Systemingenieur entwickelt) erfolgt. Damit können diese Prüfsummenwerte (Integritätsmessungen) als ein Ausgangspunkt bzw. eine Referenz zum Vergleichen der tatsächlichen Konfiguration des Endpunkts zu einem späteren Zeitpunkt verwendet werden. Somit sollten, wenn ein bestimmter Endpunkt zu einem späteren Zeitpunkt Integritätsmessungen seiner eigenen Konfiguration erzeugt (Prüfsummen berechnet), die erzeugten Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn die später erzeugten Integritätsmessungen nicht mit den entsprechenden Referenzwerten übereinstimmen, besteht die Möglichkeit, dass mit dem Endpunkt etwas passiert ist, was dessen Integrität kompromittiert hat, weil sein Konfigurationsstatus nicht wie erwartet oder ursprünglich definiert ist. Zusätzlich oder alternativ kann der Referenzwertgenerator 406 basierend auf den Konfigurationsdaten Referenzregeln entsprechend dem Status der Hardware (Peripherieeinrichtungen) erzeugen, die jedem Endpunkt zugeordnet ist.

[0031] In dem dargestellten Beispiel aus **Fig.** 4 verfügt das CM 226 über die beispielhafte Kommunika-

tionsschnittstelle 402 zum Ermöglichen der Kommunikation zwischen dem CM 226 und den anderen Endpunkten in dem Steuerungssystem 200. In einigen Beispielen wird die Konfigurationsdatei, sobald sie erzeugt wurde, über die Kommunikationsschnittstelle 410 zu jedem Endpunkt übertragen, um die Konfiguration jeder Einrichtung entsprechend den Konstruktionsvorgaben des Systemingenieurs zu ermöglichen. Des Weiteren wird, in einigen Beispielen, die Kommunikationsschnittstelle 410 verwendet, um die erzeugten Referenzwerte an das Systemintegritätsüberwachungsmodul (System Integrity Monitor - SIM) 228 der primären Arbeitsstation 220 bereitzustellen, um sie mit den zu einem späteren Zeitpunkt erhaltenen Integritätsmessungen von Endpunkten zu vergleichen.

[0032] Während in **Fig.** 4 eine beispielhafte Implementierungsweise des CM 226 aus **Fig.** 2 dargestellt ist, können ein oder mehrere der in **Fig.** 4 dargestellten Elemente. Prozesse und/oder Einrichtungen kombiniert, aufgeteilt, neu angeordnet, ausgelassen, eliminiert und/oder in jeder anderen Weise implementiert werden. Des Weiteren kann die Implementierung der beispielhaften Benutzerschnittstelle 402, des beispielhaften Konfigurationsdateigenerators 404, des beispielhaften Referenzwertgenerators 406, der beispielhaften Konfigurationsdatenbank 408, der beispielhaften Kommunikationsschnittstelle 410 und/oder, noch allgemeiner, des beispielhaften CM 226 aus **Fig.** 4 durch Hardware, Software, Firmware und/oder jede Kombination aus Hardware, Software und/oder Firmware erfolgen. Damit könnte zum Beispiel die Implementierung von jedem der beispielhaften Benutzerschnittstelle 402, des beispielhaften Konfigurationsdateigenerators 404, des beispielhaften Referenzwertgenerators 406, der beispielhaften Konfigurationsdatenbank 408, der beispielhaften Kommunikationsschnittstelle 410 und/oder, noch allgemeiner, des beispielhaften CM 226 durch eine oder mehrere Analog- oder Digitalschaltungen, Logikschaltungen, programmierbare Prozessoren, anwendungsspezifische integrierte Schaltungen (Application-Specific Integrated Circuits - ASICs), programmierbare Logikbausteine (Programmable Logic Devices - PLDs) und/oder feldprogrammierbare Logikbausteine (Field Programmable Logic Devices FLDs) erfolgen. Sollten sich einige der Vorrichtungs- oder Systemansprüche des vorliegenden Patents so lesen, als sei damit eine reine Software- und/oder Firmware-Implementierung abgedeckt, so soll hiermit mindestens eines der beispielhaften Benutzerschnittstelle 402, des beispielhaften Konfigurationsdateigenerators 404, des beispielhaften Referenzwertgenerators 406, der beispielhaften Konfigurationsdatenbank 408 und/oder der beispielhaften Kommunikationsschnittstelle 410 ausdrücklich so definiert werden, dass eine greifbare computerlesbare Speichereinrichtung oder Speicherplatte wie z. B. ein Speicher, eine Digital Versatile Disk (DVD), eine Compact Disk (CD), eine Blu-ray Disk usw. zum Speichern der Software und/oder Firmware enthalten ist. Im noch weiteren Sinne kann das beispielhafte CM 226 aus **Fig.** 2 ein oder mehrere Elemente, Prozesse und/oder Einrichtungen zusätzlich zu den oder anstelle der in **Fig.** 4 dargestellten enthalten und/oder kann mehr als eines von allen der dargestellten Elemente, Prozesse und Einrichtungen enthalten.

[0033] Fig. 5 stellt eine beispielhafte Implementierungsweise eines beispielhaften Integritätsmessungsmoduls (IMM) 230 aus **Fig.** 2 und/oder 3 dar. In dem dargestellten Beispiel umfasst das IMM 230 einen beispielhaften Integritätsmessungsgenerator 502, eine beispielhafte Integritätsmessungs-Selbsttesteinrichtung 504, eine beispielhafte Integritätsmessungs-Steuerungseinrichtung 506, einen beispielhaften Kommunikationsauthentifikator 508, eine beispielhafte Kommunikationsschnittstelle 510, ein beispielhaftes Software-Messungsregister 512, ein beispielhaftes Konfigurationsmessungsregister 514, ein beispielhaftes Hardware-Messungsregister 516, eine beispielhafte Autorisierungsinformationsdatenbank 518 und eine beispielhafte Konfigurationsdatenbank 520.

[0034] In dem dargestellten Beispiel aus **Fig.** 5 ist das IMM 230 mit dem beispielhaften Integritätsmessungsgenerator 502 ausgestattet, um Integritätsmessungen zu berechnen oder zu erzeugen, die dem konkreten Endpunkt zugeordnet sind, der das IMM 230 implementiert. In einigen Beispielen entsprechen die Integritätsmessungen der Software und/oder Firmware des Endpunkts, der das IMM 230 implementiert, und werden hier als Software-Integritätsmessungen bezeichnet. In einigen Beispielen erfolgt die Erzeugung der Software-Integritätsmessungen während der Boot-Zeit, wenn der Software-Stack des entsprechenden Endpunkts geladen wird. In einigen solcher Beispiele wird beim Laden von jedem Software-Bestandteil durch den Integritätsmessungsgenerator 502 eine kryptografische Prüfsumme für die Software berechnet und zum beispielhaften Software-Messungsregister 512 hinzugefügt (z. B. an zuvor berechnete Prüfsummenwerte angehängt). In einigen Beispielen kann, sobald die gesamte Software geladen wurde (oder bevor die gesamte Software geladen wurde), die resultierende Zeichenfolge der berechneten Prüfsummen durch einen Hashing-Algorithmus geschickt werden, um die Verwaltung der Software-Integritätsmessung zu erleichtern (z. B. Verwendung von weniger Speicher). In einigen Beispielen ist das IMM 230 als eine Hardware-Einrichtung implementiert (z. B. als ein Trusted Platform Module), um unabhängig von der restlichen durch den Endpunkt implementierten Software zu sein. Alternativ kann, in einigen Beispielen, wo keine getrennte Sicherheitseinrichtungen verfügbar ist (z. B. in einer bereits vorhandenen Ein-

richtung) das IMM 230 als Software implementiert sein, die auf den entsprechenden Endpunkt geladen wird.

[0035] In einigen Beispielen können die Integritätsmessungen alternativ oder zusätzlich der Konfiguration des Endpunkts entsprechen, der das IMM 230 implementiert, und werden hier als Konfigurationsintegritätsmessungen bezeichnet. Wie oben beschrieben kann ein Systemingenieur oder ein anderer Mitarbeiter in der Konfigurationsarbeitsstation 218 eine Konfigurationsdatei erstellen, die dann jedem Endpunkt in dem Steuerungssystem 200 bereitgestellt wird. In einigen Beispielen wird der Abschnitt der Konfigurationsdatei für das System, der für jeden konkreten Endpunkt relevant ist, zum Konfigurieren dieses Endpunkts verwendet. In einigen Beispielen werden, sobald der Endpunkt konfiguriert wurde, die Parameter, die Logik und/oder andere Konfigurationsdaten, mit denen der Endpunkt konfiguriert wurde, in der beispielhaften Konfigurationsdatenbank 520 des IMM 230 gespeichert. Damit wird, in einigen Beispielen, eine Konfigurationsintegritätsmessung erzeugt, indem eine kryptografische Prüfsumme von jedem Teil der Konfigurationsdaten in der Konfigurationsdatenbank 520 berechnet wird, der dem entsprechenden Endpunkt zugewiesen ist. Nachdem die Prüfsumme für jeden Teil der Konfigurationsdaten durch den Integritätsmessungsgenerator 502 berechnet wurde, wird die Prüfsumme an die zuvor berechneten Prüfsummen im entsprechenden Konfigurationsmessungsregister 514 angehängt. In einigen Beispielen wird eine einzige Prüfsumme für die gesamte dem Endpunkt zugewiesene Konfigurationsdatei berechnet. In anderen Beispielen werden mehrere Prüfsummen für verschiedene Abschnitte der Konfigurationsdaten berechnet und dann kombiniert (z. B. in einer Zeichenfolge aneinandergehängt). In einigen Beispielen können getrennte Konfigurationsdateien für getrennte Ebenen im Software-Stack gelten, und/oder der Endpunkt kann anderweitig mehrere Konfigurationsdateien speichern (z. B. verschiedene Versionen). In einigen Beispielen wird eine getrennte Prüfsumme für die Konfigurationsdaten berechnet, die jeder Konfigurationsdatei zugeordnet sind, und dann werden die resultierenden Werte aneinandergehängt. In einigen Beispielen kann, sobald eine Prüfsumme für alle Konfigurationsdaten berechnet wurde (oder bevor alle Konfigurationsdaten analysiert wurden), die resultierende Zeichenfolge der berechneten Prüfsummen durch einen Hashing-Algorithmus geschickt werden, um die Verwaltung der Konfigurationsintegritätsmessung zu erleichtern.

[0036] In einigen Beispielen können die Integritätsmessungen alternativ oder zusätzlich der Hardware entsprechen, die dem Endpunkt zugeordnet ist, der das IMM 230 implementiert, und werden hier als Hardware-Integritätsmessungen bezeichnet. In einigen Beispielen entspricht die einem Endpunkt zugeordnete Hardware den Peripherieeinrichtungen, die in den Endpunkt eingesteckt sind oder anderweitig mit diesem gekoppelt sind (z. B. in eine Steuerungseinrichtung eingesteckte E/A-Karten). In einigen Beispielen wird eine Hardware-Integritätsmessung erzeugt, indem eine kryptografische Prüfsumme für jede dem Endpunkt zugeordnete Peripherieeinrichtung berechnet wird. In einigen Beispielen werden die Peripherieeinrichtungen, für die Messungen (z. B. Prüfsummen) erzeugt werden sollen, basierend auf den Konfigurationsdaten identifiziert, die in der Konfigurationsdatenbank 520 gespeichert sind. In einigen Beispielen erfolgt die Erzeugung der Integritätsmessungen basierend auf der Konfigurationsdatei, die der Peripherieeinrichtung zugeordnet ist. In anderen Beispielen stellt die Peripherieeinrichtung als Reaktion auf eine Anforderung von dem Endpunkt einen Prüfsummenwert bereit, der aussagekräftig in Bezug auf ihren Status ist. Nachdem die Prüfsumme für jede Peripherieeinrichtung durch den Integritätsmessungsgenerator 502 berechnet wurde, wird jede Prüfsumme an die zuvor berechneten Prüfsummen im entsprechenden Hardware-Messungsregister 516 angehängt. In einigen Beispielen kann, sobald die Prüfsummen für die gesamte (oder einen Teil der) Hardware berechnet wurde, die einem Endpunkt zugeordnet ist, die resultierende Zeichenfolge der berechneten Prüfsummen durch einen Hashing-Algorithmus geschickt werden, um die Verwaltung der Hardware-Integritätsmessung zu erleichtern.

[0037] In dem dargestellten Beispiel aus **Fig.** 5 ist das IMM 230 mit der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 504 ausgestattet, um Vorabtests der Funktionen durchzuführen, die zum Erzeugen der Integritätsmessungen genutzt werden. Das heißt, in einigen Beispielen führt die Integritätsmessungs-Selbsttesteinrichtung 504 einen internen Test der Fähigkeit des Integritätsmessungsgenerators 502 zur ordnungsgemäßen Berechnung der Werte entsprechend den Integritätsmessungen durch. Zum Beispiel kann die Integritätsmessungs-Selbsttesteinrichtung 504 verifizieren, dass die durch den Integritätsmessungsgenerator 502 implementierten Hash-Algorithmen und Prüfsummenalgorithmen basierend auf bekannten Eingaben die erwarteten Ausgaben erzeugen.

[0038] In dem dargestellten Beispiel aus **Fig.** 5 ist das IMM 230 mit der beispielhaften Integritätsmessungs-Steuerungseinrichtung 506 ausgestattet, um verschiedene Operationen des IMM 230 zu steuern. In einigen Beispielen kommuniziert die Integritätsmessungs-Steuerungseinrichtung 506 Anweisungen oder Befehle zu anderen Abschnitten des beispielhaften IMM 230, um die Operationen dieser Abschnitte zu steuern. Zum Beispiel weist die Integritätsmessungs-Steuerungseinrichtung 506 an,

wann Integritätsmessungen durch den Integritätsmessungsgenerator 502 erzeugt und die Werte zu den jeweiligen Registern 512, 514, 516 hinzugefügt werden sollen; wann die Integritätsmessungen über die Kommunikationsschnittstelle 510 gemeldet werden sollen; wann die gespeicherten Integritätsmessungen aus den Registern 512, 514, 516 gelöscht werden sollen und so weiter.

[0039] In dem dargestellten Beispiel aus **Fig.** 5 ist das IMM 230 mit dem beispielhaften Kommunikationsauthentifikator 508 ausgestattet, um die Kommunikation mit anderen Endpunkten zu authentifizieren und/oder zu verifizieren, nachdem es vollständigen Kommunikationszugriff auf das entsprechende Steuerungsnetzwerk erhalten hat. In einigen Beispielen wird der vollständige Kommunikationszugriff durch das SIM 228 ermöglicht, das Autorisierungsinformationen bereitstellt, die durch den Endpunkt verwendet werden, um geheime und/oder öffentliche Werte für digitale Signaturen, Verschlüsselung/Entschlüsselung, Authentifizierung und so weiter zu erzeugen.

[0040] In einigen Beispielen werden die durch den Kommunikationsauthentifikator 508 verwendeten Autorisierungsinformationen an das IMM 230 bereitgestellt, um den vollständigen Kommunikationszugriff zu ermöglichen, nachdem die Integrität des Endpunkts bestätigt oder verifiziert wurde. Zum Beispiel bildet einen Teil der erstmaligen Konfiguration und Inbetriebnahme eines Endpunkts beim Empfangen einer neuen Konfigurationsdatei den Neustart (Rebooting) des Endpunkts. Während des Neustarts erzeugt, in einigen Beispielen, der Integritätsmessungsgenerator 502 Software-Integritätsmessungen wie oben beschrieben. In einigen Beispielen können auch Konfigurations- und/oder Hardware-Integritätsmessungen erzeugt werden. Diese Integritätsmessungen sind aussagekräftig in Bezug auf den aktuellen Status des Endpunkts (d. h. zum Zeitpunkt des Neustarts). Nachdem er entsprechend der Konfigurationsdatei konfiguriert wurde, kann der neu in Betrieb genommene Endpunkt die Registrierung in dem und die Zulassung für das Steuerungssystem anfordern. Bevor die Zulassung zur Kommunikation über das Steuerungsnetzwerk erteilt wird, werden die Integritätsmessungen mit Referenzwerten verglichen. Wenn die Integritätsmessungen mit den Referenzwerten übereinstimmen, erhält der Endpunkt vollständigen Kommunikationszugriff, indem ihm die Autorisierungsinformationen bereitgestellt werden, aus denen die geheimen und/oder zugehörigen öffentlichen Werte erzeugt werden. In einigen Beispielen werden die Autorisierungsinformationen anderen Endpunkten in dem entsprechenden Netzwerk bereitgestellt, sodass alle Endpunkte mit dem neu zugelassenen Endpunkt (sowie miteinander) kommunizieren können.

[0041] Es besteht die Möglichkeit, dass ein bestimmter Endpunkt nach der erstmaligen Konfiguration und nach dem Empfang der Autorisierungsinformationen, die dem Endpunkt vollen Kommunikationszugriff gewähren, auf die eine oder andere Weise kompromittiert wird. Dementsprechend werden, in einigen Beispielen, die Integritäten der Endpunkte im Lauf der Zeit überwacht oder zu verschiedenen Zeitpunkten verifiziert, um etwaige Änderungen im Status der Endpunkte zu erkennen, die aussagekräftig in Bezug auf einen Integritätsverlust sind (z. B. dass er sich nicht mehr in einem vertrauenswürdigen Status befindet). In einigen Beispielen werden, sobald ein kompromittierter Endpunkt erkannt wurde, neue Autorisierungsinformationen an alle anderen Endpunkte verteilt, die für die zukünftige Konfiguration verwendet werden müssen. Auf diese Weise wird dem kompromittierten Endpunkt der vollständige Kommunikationszugriff für die Zukunft verweigert, da die zuvor empfangenen Autorisierungsinformationen nicht mehr gültig sind. Des Weiteren werden, in einigen solcher Beispiele, eindeutige Autorisierungsinformationen an den kompromittierten Endpunkt bereitgestellt, um den Endpunkt in einen Abhilfemodus zu versetzen und/oder um einen eingeschränkten Kommunikationszugriff zu ermöglichen.

[0042] In einigen Beispielen wird das Problem, dass ein Endpunkt mit verdächtiger Integrität zuvor bereitgestellte Autorisierungsinformationen verwendet, dadurch gelöst, dass der Kommunikationsauthentifikator 508 die Autorisierungsinformationen beim erstmaligen Empfang in einer Weise verschlüsselt, dass sie nachfolgend nur entschlüsselt und verwendet werden können, wenn sich der Endpunkt im selben Status wie beim erstmaligen Empfang der Autorisierungsinformationen befindet. Damit ist, wenn ein Endpunkt auf die eine oder andere Weise kompromittiert wird, die sich auf seinen aktuellen Status auswirkt (angezeigt durch eine Änderung in den Integritätsmessungen) der Endpunkt nicht mehr in der Lage, auf die Autorisierungsinformationen zuzugreifen (diese zu entschlüsseln), die zur Kommunikation mit anderen Endpunkten erforderlich sind. Im Ergebnis dessen kann, in einigen Beispielen, die Kommunikation von einem Endpunkt, der kompromittiert wurde, verhindert werden, ohne dass das SIM 228 den Integritätsverlust des Endpunkts erkennen muss.

[0043] Einige durch einen Endpunkt gesendete Meldungen enthalten Berichte zu den Integritätsmessungen für den Endpunkt. Eine Herausforderung in der Netzwerksicherheit stellt das Problem des lügenden Endpunkts dar. Zum Beispiel in dem Fall, dass die Integrität eines Endpunkts verifiziert ist, indem die Integritätsmessungen mit Referenzwerten entsprechend einem bekannten vertrauenswürdigen Status verglichen wurden, besteht die Möglichkeit, dass ein

infizierter Endpunkt lediglich zuvor ermittelte Integritätsmessungen wiederholt, die mit den Referenzwerten übereinstimmen, anstatt Integritätsmessungen bereitzustellen, die aus dem tatsächlichem und/oder aktuellen Status des Endpunkts erzeugt wurden (die anzeigen könnten, dass der Endpunkt infiziert ist). Um dieses Problem zu lösen, wird dem IMM 230, in einigen Beispielen, bei jeder Anforderung der Integritätsmessungen eine Nonce (ein eindeutiger Wert für eine einmalige Verwendung) bereitgestellt. In einigen dieser Beispiele kombiniert der Kommunikationsauthentifikator 508 die Nonce mit der Integritätsmessung, wenn er auf die Anforderung reagiert, sodass ein Endpunkt erkannt werden kann, der lediglich bereits vorhandene Werte für die Integritätsmessung berichtet (ohne die Nonce).

[0044] Während in **Fig.** 5 eine beispielhafte Implementierungsweise des IMM 230 aus **Fig.** 2 und/oder 3 dargestellt ist, können ein oder mehrere der in **Fig.** 5 dargestellten Elemente. Prozesse und/oder Einrichtungen kombiniert, aufgeteilt, neu angeordnet, ausgelassen, eliminiert und/oder in jeder anderen Weise implementiert werden. Des Weiteren kann die Implementierung des beispielhaften Integritätsmessungsgenerators 502, der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 504, der beispielhaften Integritätsmessungs-Steuerungseinrichtung 506, des beispielhaften Kommunikationsauthentifikators 508, der beispielhaften Kommunikationsschnittstelle 510, des beispielhaften Software-Messungsregisters 512, des beispielhaften Konfigurationsmessungsregisters 514, des beispielhaften Hardware-Messungsregisters 516, der beispielhaften Autorisierungsinformationsdatenbank 518, der beispielhaften Konfigurationsdatenbank 520 und/oder, noch allgemeiner, des beispielhaften IMM 230 aus **Fig.** 5 durch Hardware, Software, Firmware und/oder jede Kombination aus Hardware, Software und/oder Firmware erfolgen. Damit könnte zum Beispiel die Implementierung von jedem des beispielhaften Integritätsmessungsgenerators 502, der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 504, der beispielhaften Integritätsmessungs-Steuerungseinrichtung 506, des beispielhaften Kommunikationsauthentifikators 508, der beispielhaften Kommunikationsschnittstelle 510, des beispielhaften Software-Messungsregisters 512, des beispielhaften Konfigurationsmessungsregisters 514, des beispielhaften Hardware-Messungsregisters 516, der beispielhaften Autorisierungsinformationsdatenbank 518, der beispielhaften Konfigurationsdatenbank 520 und/oder, noch allgemeiner, des beispielhaften IMM 230 durch eine oder mehrere Analog- oder Digitalschaltungen, Logikschaltungen, programmierbare Prozessoren, anwendungsspezifische integrierte Schaltungen (Application-Specific Integrated Circuits - ASICs), programmierbare Logikbausteine (Programmable Logic Devices - PLDs) und/oder feldprogrammierbare Logikbausteine (Field Programmable Logic Devices FLDs) erfolgen. Sollten sich einige der Vorrichtungs- oder Systemansprüche des vorliegenden Patents so lesen, als sei damit eine reine Software- und/oder Firmware-Implementierung abgedeckt, so soll hiermit mindestens eines des beispielhaften Integritätsmessungsgenerators 502, der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 504, der beispielhaften Integritätsmessungs-Steuerungseinrichtung 506, des beispielhaften Kommunikationsauthentifikators 508, der beispielhaften Kommunikationsschnittstelle 510, des beispielhaften Software-Messungsregisters 512, des beispielhaften Konfigurationsmessungsregisters 514, des beispielhaften Hardware-Messungsregisters 516, der beispielhaften Autorisierungsinformationsdatenbank 518 und/oder der beispielhaften Konfigurationsdatenbank 520 ausdrücklich so definiert werden, dass eine greifbare computerlesbare Speichereinrichtung oder Speicherplatte wie z. B. ein Speicher, eine Digital Versatile Disk (DVD), eine Compact Disk (CD), eine Blu-ray Disk usw. zum Speichern der Software und/oder Firmware enthalten ist. Im noch weiteren Sinne kann das beispielhafte IMM 230 aus **Fig.** 2 und/oder 3 ein oder mehrere Elemente, Prozesse und/oder Einrichtungen zusätzlich zu den oder anstelle der in **Fig.** 5 dargestellten enthalten und/oder kann mehr als eines von allen der dargestellten Elemente, Prozesse und Einrichtungen enthalten.

[0045] **Fig.** 6 stellt eine beispielhafte Implementierungsweise der beispielhaften Systemintegritätsüberwachungsmodule (System Integrity Monitor - SIM) 228 aus **Fig.** 2 und/oder 3 dar. In dem dargestellten Beispiel umfasst das SIM 228 eine beispielhafte Kommunikationsschnittstelle 602, einen beispielhaften Integritätsmessungskomparator 604, eine beispielhafte Autorisierungssteuerungseinrichtung 606, einen beispielhaften Integritätsmessungsgenerator 608, eine beispielhafte Integritätsmessungs-Selbsttesteinrichtung 610, eine beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, eine beispielhafte Benutzerschnittstelle 614, eine beispielhafte Referenzwertdatenbank 616, eine beispielhafte Integritätsmessungsdatenbank 618, eine beispielhafte Autorisierungsinformationsdatenbank 620, ein beispielhaftes Software-Messungsregister 622, ein beispielhaftes Hardware-Messungsregister 624 und ein beispielhaftes Konfigurationsmessungsregister 626.

[0046] In dem dargestellten Beispiel aus **Fig.** 6 verfügt das SIM 228 über die beispielhafte Kommunikationsschnittstelle 602 zum Ermöglichen der Kommunikation mit anderen Endpunkten in dem entsprechenden Netzwerk. Zum Beispiel sendet die Kommunikationsschnittstelle 602 des SIM 228 Anforderungen für Integritätsmessungen von anderen Endpunkten aus. In gleicher Weise empfängt die beispielhafte Kommunikationsschnittstelle 602 Berichte

mit Integritätsmessungen von den IMMs 230 der anderen Endpunkte und/oder von dem SIM 228 einer primären Arbeitsstation eines Subsystems (z. B. des Steuerungssystems 200 als ein Endpunkt in dem Beaufsichtigungssystem 300). Damit kann, in einigen Beispielen, das SIM 228 einer primären Arbeitsstation eines Steuerungssystems die Integritätsmessungen an ein SIM 228 einer primären Arbeitsstation eines Beaufsichtigungssystems kommunizieren.

[0047] Das beispielhafte SIM 228 aus **Fig.** 6 verfügt über den beispielhaften Integritätsmessungskomparator 604 zum Vergleichen der Integritätsmessungen mit Referenzwerten, die in der beispielhaften Referenzwertdatenbank 616 gespeichert sind. In einigen Beispielen werden die Referenzwerte basierend auf der Konfigurationsdatei empfangen und/oder erzeugt, die mit dem Konfigurationsmodul 226 der Konfigurationsarbeitsstation 218 erstellt wurde. In einigen Beispielen werden die Referenzwerte von einem Benutzer empfangen, der die Werte manuell über die beispielhafte Benutzerschnittstelle 614 basierend auf Werten eingibt, die durch einen Originalhersteller des entsprechenden Endpunkts bereitgestellt wurden (z. B. enthalten in der Dokumentation, die mit der Einrichtung ausgeliefert wurde und/oder online bereitgestellt wurde). Auf diese Weise sind die Werte vertrauenswürdig zum Zeitpunkt der Herstellung, sodass jede Manipulation an den Einrichtungen während des Versands erkannt werden kann. Zusätzlich oder alternativ basieren, in einigen Beispielen, einige oder alle der Referenzwerte auf Integritätsmessungen, die durch die IMMs 230 der jeweiligen Endpunkte erzeugt wurden, als die Endpunkte erstmalig konfiguriert, in Betrieb genommen und/oder zum entsprechenden Netzwerk hinzugefügt wurden. In einigen solcher Beispiele bestätigt ein Ingenieur und/oder anderer Benutzer die Verwendung solcher Integritätsmessungen als Referenzwerte. Auf diese Weise hat der Ingenieur eine Möglichkeit zu verifizieren, dass jeder konkrete Endpunkt korrekt konfiguriert ist und erwartungsgemäß funktioniert, wenn diese Werte zu späteren Zeitpunkten dem entsprechen, was beim Testen der Integrität des Endpunkts verwendet wurde.

[0048] In dem dargestellten Beispiel aus **Fig.** 6 ist das SIM 228 mit der beispielhaften Autorisierungssteuerungseinrichtung 606 ausgestattet, um basierend auf den Ergebnissen des Vergleichs der berichteten Integritätsmessungen mit den gespeicherten Referenzwerten die geeignete Kommunikationsautorisierung bzw. den geeigneten Kommunikationszugriff für jeden Endpunkt festzulegen. In einigen Beispielen stellt die Autorisierungssteuerungseinrichtung 606 bei Übereinstimmung der Integritätsmessungen mit den Referenzwerten Autorisierungsinformationen an den Endpunkt und alle anderen Endpunkte bereit, um die Kommunikation zwischen diesen zu ermöglichen. In einigen Beispielen werden die an die anderen Endpunkte verteilten Autorisierungsinformationen zu nachfolgenden Referenzzwecken auch in der beispielhaften Autorisierungsinformationsdatenbank 620 gespeichert.

[0049] In einigen Beispielen ergreift die Autorisierungssteuerungseinrichtung 606 bei Nichtübereinstimmung der Integritätsmessungen mit den entsprechenden Referenzwerten Abhilfemaßnahmen. In einigen Beispielen umfassen die Abhilfemaßnahmen das Ablehnen des Kommunikationszugriffs für den Endpunkt (z. B. durch Bereitstellung neuer Autorisierungsinformationen an die anderen Endpunkte). In anderen Beispielen wird der Kommunikationszugriff für den Endpunkt nicht vollständig abgelehnt, sondern auf bestimmte Weise eingeschränkt. Zum Beispiel können dem Endpunkt eindeutige Autorisierungsinformationen bereitgestellt werden, die den Endpunkt auf einen Kommunikationszugriff für Firmware-Updates einschränken (z. B. ist dann die einzige Kommunikation, die durchgeführt werden kann, mit dem Aktualisieren der Firmware des Endpunkts verknüpft). In einigen Beispielen können dem Endpunkt eindeutige Autorisierungsinformationen bereitgestellt werden, die den Endpunkt auf einen Kommunikationszugriff zur Konfiguration einschränken (z. B. ist dann die einzige Kommunikation, die durchgeführt werden kann, mit dem Aktualisieren der Konfiguration des Endpunkts verknüpft). In einigen Beispielen kann die Kommunikation von dem Endpunkt zwar gestattet werden, diese Kommunikation wird aber markiert. In einigen Beispielen löst die Autorisierungssteuerungseinrichtung 606 einen Alarm und/oder eine Warnung aus, um einen Benutzer über einen potenziellen Integritätsverlust zu informieren. In einigen Beispielen stellt die Autorisierungssteuerungseinrichtung 606 zusätzlich zur Verhinderung der Kommunikation von einem korrumpierten Endpunkt Autorisierungsinformationen an einen redundanten oder der Ausfallsicherung dienenden Endpunkt bereit, der die Rolle des korrumpierten Endpunkts übernehmen soll. In einigen Beispielen kann die Autorisierungssteuerungseinrichtung 606 eine beliebige Kombination der oben erwähnten Abhilfemaßnahmen implementieren.

[0050] Zusätzlich erzeugt, in einigen Beispielen, die Autorisierungssteuerungseinrichtung 606 Nonce-Werte, die zusammen mit den Anforderungen für Integritätsmessungen bereitgestellt werden, die zur Verwendung durch das entsprechende IMM 230 beim Reagieren verwendet werden sollen, sodass die Reaktionen von den Endpunkten als legitim verifiziert werden können.

[0051] In dem dargestellten Beispiel aus **Fig.** 6 ist das SIM 228 mit dem beispielhaften Integritätsmessungsgenerator 608 ausgestattet, um aussagekräf-

tige Integritätsmessungen in Bezug auf den Status der Integrität des gesamten Systems von Endpunkten zu erzeugen, die von dem SIM 228 überwacht werden. In einigen Beispielen erfolgt die Erzeugung solcher systemweiten Integritätsmessungen durch das Kombinieren der Integritätsmessungen, die von jedem Endpunkt in dem zugehörigen Netzwerk erfasst werden, das von dem SIM 228 überwacht wird. Diese kombinierte Integritätsmessung kann an ein anderes SIM 228 auf einer Arbeitsstation berichtet werden, die einer höheren Ebene in der Hierarchie 100 aus **Fig.** 1 zugeordnet ist. In einigen Beispielen erzeugt der Integritätsmessungsgenerator 608 die kombinierten Integritätsmessungen durch Anhängen jeder von jedem Endpunkt in dem Netzwerk berichteten Integritätsmessung in den entsprechenden Software-, Hardware- und Konfigurationsregistern 622, 624, 626 und anschließendes Berechnen einer Prüfsumme für die resultierende Zeichenfolge aus Werten. In einigen Beispielen fordert das SIM 228 die Integritätsmessungen von jedem Endpunkt an, bevor die kombinierten Integritätsmessungen erzeugt werden. Zusätzlich oder alternativ können die kombinierten Integritätsmessungen basierend auf zuvor berichteten Integritätsmessungen erzeugt werden, die in der Integritätsmessungsdatenbank 618 gespeichert sind.

[0052] In dem dargestellten Beispiel aus **Fig.** 6 ist das SIM 228 mit der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 610 ausgestattet, um Vorabtests der Funktionen durchzuführen, die zum Erzeugen der Integritätsmessungen genutzt werden. Das heißt, in einigen Beispielen führt die Integritätsmessungs-Selbsttesteinrichtung 610 einen internen Test der Fähigkeit des Integritätsmessungsgenerators 608 zur ordnungsgemäßen Berechnung der Werte (z. B. Prüfsummen) entsprechend den Integritätsmessungen durch. Zum Beispiel kann die Integritätsmessungs-Selbsttesteinrichtung 610 verifizieren, dass die durch den Integritätsmessungsgenerator 608 implementierten Hash-Algorithmen und Prüfsummenalgorithmen basierend auf bekannten Eingaben die erwarteten Ausgaben erzeugen.

[0053] In dem dargestellten Beispiel aus **Fig.** 6 ist das SIM 228 mit der beispielhaften Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ausgestattet, um verschiedene Operationen des SIM 228 zu steuern. In einigen Beispielen kommuniziert die Integritätsüberwachungsmodul-Steuerungseinrichtung 612 Anweisungen oder Befehle zu anderen Abschnitten des beispielhaften SIM 228, um die Operationen dieser Abschnitte zu steuern. Zum Beispiel kann die Integritätsüberwachungsmodul-Steuerungseinrichtung 612 einen Zeitplan führen, wann die Integritätsmessungen von den IMMs 230 von anderen Endpunkten angefordert werden sollen. In einigen Beispielen steuert und/oder definiert die Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die Reihenfolge, in der die verschiedenen Integritätsmessungen von den verschiedenen Endpunkten kombiniert werden, um die kombinierten Integritätsmessungen zu erzeugen, weil die Reihenfolge Einfluss auf den resultierenden Prüfsummenwert der gesamten Zeichenfolge hat.

[0054] Während in **Fig.** 6 eine beispielhafte Implementierungsweise des SIM 228 aus **Fig.** 2 und/oder 3 dargestellt ist, können ein oder mehrere der in **Fig.** 6 dargestellten Elemente. Prozesse und/oder Einrichtungen kombiniert, aufgeteilt, neu angeordnet, ausgelassen, eliminiert und/oder in jeder anderen Weise implementiert werden. Des Weiteren kann die Implementierung der beispielhaften Kommunikationsschnittstelle 602, des beispielhaften Integritätsmessungskomparators 604, der beispielhaften Autorisierungssteuerungseinrichtung 606, des beispielhaften Integritätsmessungsgenerators 608, der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 610, der beispielhaften Integritätsüberwachungsmodul-Steuerungseinrichtung 612, der beispielhaften Benutzerschnittstelle 614, der beispielhaften Referenzwertdatenbank 616, der beispielhaften Integritätsmessungsdatenbank 618, der beispielhaften Autorisierungsinformationsdatenbank 620, des beispielhaften Software-Messungsregisters 622, des beispielhaften Hardware-Messungsregisters 624, des beispielhaften Konfigurationsmessungsregisters 626 und/oder, noch allgemeiner, des beispielhaften SIM 228 aus **Fig.** 6 durch Hardware, Software, Firmware und/oder jede Kombination aus Hardware, Software und/oder Firmware erfolgen. Damit könnte zum Beispiel die Implementierung von jedem der beispielhaften Kommunikationsschnittstelle 602, des beispielhaften Integritätsmessungskomparators 604, der beispielhaften Autorisierungssteuerungseinrichtung 606, des beispielhaften Integritätsmessungsgenerators 608, der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 610, der beispielhaften Integritätsüberwachungsmodul-Steuerungseinrichtung 612, der beispielhaften Benutzerschnittstelle 614, der beispielhaften Referenzwertdatenbank 616, der beispielhaften Integritätsmessungsdatenbank 618, der beispielhaften Autorisierungsinformationsdatenbank 620, des beispielhaften Software-Messungsregisters 622, des beispielhaften Hardware-Messungsregisters 624, des beispielhaften Konfigurationsmessungsregisters 626 und/oder, noch allgemeiner, des beispielhaften SIM 228 durch eine oder mehrere Analog- oder Digitalschaltungen, Logikschaltungen, programmierbare Prozessoren, anwendungsspezifische integrierte Schaltungen (Application-Specific Integrated Circuits - ASICs), programmierbare Logikbausteine (Programmable Logic Devices - PLDs) und/oder feldprogrammierbare Logikbausteine (Field Programmable Logic Devices FLDs) erfolgen. Sollten sich einige der Vorrichtungs- oder Systemansprüche des vorliegenden Patents so lesen, als sei damit eine reine

Software- und/oder Firmware-Implementierung abgedeckt, so soll hiermit mindestens eines der beispielhaften Kommunikationsschnittstelle 602, des beispielhaften Integritätsmessungskomparators 604, der beispielhaften Autorisierungssteuerungseinrichtung 606, des beispielhaften Integritätsmessungsgenerators 608, der beispielhaften Integritätsmessungs-Selbsttesteinrichtung 610, der beispielhaften Integritätsüberwachungsmodul-Steuerungseinrichtung 612, der beispielhaften Benutzerschnittstelle 614, der beispielhaften Referenzwertdatenbank 616, der beispielhaften Integritätsmessungsdatenbank 618, der beispielhaften Autorisierungsinformationsdatenbank 620, des beispielhaften Software-Messungsregisters 622, des beispielhaften Hardware-Messungsregisters 624 und/oder des beispielhaften Konfigurationsmessungsregisters 626 ausdrücklich so definiert werden, dass eine greifbare computerlesbare Speichereinrichtung oder Speicherplatte wie z. B. ein Speicher, eine Digital Versatile Disk (DVD), eine Compact Disk (CD), eine Blu-ray Disk usw. zum Speichern der Software und/oder Firmware enthalten ist. Im noch weiteren Sinne kann das beispielhafte SIM 228 aus **Fig.** 2 und/oder 3 ein oder mehrere Elemente, Prozesse und/oder Einrichtungen zusätzlich zu den oder anstelle der in **Fig.** 6 dargestellten enthalten und/oder kann mehr als eines von allen der dargestellten Elemente, Prozesse und Einrichtungen enthalten.

[0055] Ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften CM 226 aus **Fig.** 2 ist in **Fig.** 7 dargestellt. Ablaufdiagramme zur Darstellung von beispielhaften Verfahren zur Implementierung des beispielhaften IMM 230 aus **Fig.** 2 und/oder 3 sind in **Fig.** 8-10 dargestellt. Ablaufdiagramme zur Darstellung von beispielhaften Verfahren zur Implementierung des beispielhaften SIM 228 aus **Fig.** 2 und/oder 3 sind in **Fig.** 11-16 dargestellt. Die Verfahren können mithilfe von maschinell lesbaren Anweisungen implementiert werden, welche Programme zur Ausführung durch einen Prozessor wie z. B. die Prozessoren 1712, 1812, 1910 bilden, die in den beispielhaften Prozessorplattformen 1700, 1800, 1900 dargestellt sind, welche im Folgenden im Zusammenhang mit **Fig.** 17-19 erläutert werden. Die Programme können in Software verkörpert sein, die auf einem greifbaren computerlesbaren Speichermedium wie z. B. einer CD-ROM, einer Diskette, einer Festplatte, einer Digital Versatile Disk (DVD), einer Compact Disk (CD), einer Blu-ray Disk oder einem mit den Prozessoren 1712, 1812, 1910 verknüpften Speicher gespeichert sind, aber die gesamten Programme und/oder Teile davon könnten alternativ durch eine andere Einrichtung als die Prozessoren 1712, 1812, 1910 ausgeführt werden und/oder in Firmware oder dedizierter Hardware verkörpert sein. Des Weiteren können, obwohl die beispielhaften Programme unter Bezug auf das in **Fig.** 7-16 dargestellte Ablaufdiagramm beschrieben werden, alternativ viele andere Verfahren zur Implementierung des bespielhaften CM 226, des beispielhaften IMM 230 und des beispielhaften SIM 228 verwendet werden. Zum Beispiel kann die Reihenfolge der Ausführung der Blöcke verändert sein, und/oder einige der beschriebenen Blöcke können verändert, eliminiert oder kombiniert sein.

[0056] Wie oben erwähnt, können die beispielhaften Verfahren aus **Fig.** 7-16 mithilfe von codierten Anweisungen (z. B. computer- und/oder maschinell lesbaren Anweisungen) implementiert werden, die auf einem greifbaren computerlesbaren Speichermedium wie z. B. einer Festplatte, einem Flash-Speicher, einem Festwertspeicher (Read-Only Memory - ROM), einer Compact Disk (CD), einer Digital Versatile Disk (DVD), einem Zwischenspeicher, einem Direktzugriffsspeicher (Random-Access Memory - RAM) und/oder jeder anderen Speichereinrichtung oder jedem anderen Speicherdatenträger gespeichert sind, auf denen Informationen für eine beliebige Zeitdauer gespeichert sind (z. B. für längere Zeiträume, dauerhaft, für kurze Momente, zum temporären Puffern und/oder zum Zwischenspeichern der Informationen). Wenn hier von einem greifbaren computerlesbaren Speichermedium die Rede ist, dann bezieht das per Definition ausdrücklich jede Art von computerlesbarer Speichereinrichtung und/oder computerlesbarem Speicherdatenträger ein und schließt das Verbreiten von Signalen aus und schließt Übertragungsmedien aus. Die hier verwendeten Begriffe „greifbares computerlesbares Speichermedium“ und „greifbares maschinell lesbares Speichermedium“ werden austauschbar verwendet. Zusätzlich oder alternativ können die beispielhaften Verfahren aus **Fig.** 7-16 mithilfe von codierten Anweisungen (z. B. computer- und/oder maschinell lesbaren Anweisungen) implementiert werden, die auf einem nicht-transitorischen computer- und/oder maschinell lesbaren Medium wie z. B. einer Festplatte, einem Flash-Speicher, einem Festwertspeicher, einer Compact Disk, einer Digital Versatile Disk, einem Zwischenspeicher, einem Direktzugriffsspeicher und/oder jeder anderen Speichereinrichtung oder jedem anderen Speicherdatenträger gespeichert sind, auf denen Informationen für eine beliebige Zeitdauer gespeichert sind (z. B. für längere Zeiträume, dauerhaft, für kurze Momente, zum temporären Puffern und/oder zum Zwischenspeichern der Informationen). Wenn hier von einem nicht-transitorischen computerlesbaren Medium die Rede ist, dann bezieht das per Definition ausdrücklich jede Art von computerlesbarer Speichereinrichtung und/oder computerlesbarem Speicherdatenträger ein und schließt das Verbreiten von Signalen aus und schließt Übertragungsmedien aus. Wenn hier der Begriff „mindestens“ als Übergangsbegriff in einem Oberbegriff eines Anspruchs verwendet wird, dann gilt dieser als offen in gleicher

Weise, wie auch der Begriff „umfassend“ als offen gilt.

[0057] Nunmehr detailliert auf die Figuren Bezug nehmend, ist **Fig.** 7 ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Konfigurationsmoduls (Configuration Module - CM) 226 aus **Fig.** 2 und/oder 4. Das beispielhafte Verfahren beginnt in Block 700, wo die beispielhafte Benutzeroberfläche Konfigurationsdaten für ein Steuerungssystem (z. B. das Steuerungssystem 200 aus **Fig.** 2) empfängt. In einigen Beispielen entsprechen die Konfigurationsdaten der Gesamtheit an Konfigurationsparametern, Logik und Eingaben, die durch einen Ingenieur bereitgestellt werden, um ein Steuerungssystem zu konfigurieren. In Block 702 empfängt die beispielhafte Benutzeroberfläche Software- und/oder Hardware-Referenzwerte für Endpunkte in dem Steuerungssystem. In einigen Beispielen entsprechen die Software-Referenzwerte und die Hardware-Referenzwerte Werten, die durch den Ingenieur basierend auf Werten eingegeben wurden, welche durch den bzw. die Hersteller der Endpunkteinrichtungen spezifiziert wurden. In einigen Beispielen werden die Software- und/oder Hardware-Werte zum selben Zeitpunkt wie die Konfigurationsdaten erfasst. In anderen Beispielen werden die Software- und/oder Hardware-Daten getrennt von den Konfigurationsdaten erfasst. In einigen Beispielen werden keine Software- und/oder Hardware-Referenzwerte erfasst. In einigen Beispielen können die Software- und/oder Hardware-Referenzwerte zu einem späteren Zeitpunkt erfasst und/oder erzeugt werden, wie das im Folgenden beschrieben ist.

[0058] In Block 704 erzeugt der beispielhafte Konfigurationsdateigenerator 404 basierend auf den Konfigurationsdaten eine Konfigurationsdatei. In Block 706 erzeugt der beispielhafte Referenzwertgenerator 406 basierend auf der Konfigurationsdatei Referenzwerte für einen Endpunkt in dem Steuerungssystem. In einigen Beispielen erfolgt die Erzeugung der Referenzwerte durch Berechnen von einer oder mehreren Prüfsummen für die Konfigurationsdatei und/oder Abschnitten davon, die dem Endpunkt zugewiesen sind. In einigen Beispielen, wo Software- und Hardware-Referenzwerte in Block 702 empfangen wurden, entspricht die erzeugte Referenz den Konfigurationsreferenzwerten für den entsprechenden Endpunkt. In einigen Beispielen enthalten die erzeugten Referenzwerte Hardware-Referenzwerte (z. B. wo solche Werte nicht in Block 702 vom Benutzer definiert werden). In Block 708 ermittelt der Konfigurationsreferenzwertgenerator 406, ob es einen weiteren Endpunkt mit entsprechenden Konfigurationsdaten gibt. Wenn das der Fall ist, kehrt die Steuerung zu Block 706 zurück, um die entsprechenden Konfigurationsreferenzwerte für den Endpunkt zu berechnen. Andernfalls wird die Steuerung mit Block 710 fortgesetzt, wo die beispielhafte Kommunikationsschnittstelle 410 die Konfigurationsdatei zu den Endpunkten des Steuerungssystems überträgt. Basierend auf dieser Übertragung kann jeder Endpunkt die Konfigurationsdatei (oder relevante Abschnitte davon) empfangen und/oder herunterladen und instantiieren und dann einen Neustart durchführen, um den Konfigurationsprozess abzuschließen, wie das nachstehend detailliert beschrieben wird. In Block 712 überträgt die beispielhafte Kommunikationsschnittstelle 410 die Referenzwerte (z. B. die Konfigurationsreferenzwerte, Software-Referenzwerte und/oder Hardware-Referenzwerte) zu einem Systemintegritätsüberwachungsmodul (System Integrity Monitor - SIM) (z. B. dem SIM 228 aus **Fig.** 2). Danach endet das beispielhafte Verfahren aus **Fig.** 7.

[0059] **Fig.** 8 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Integritätsmessungsmoduls (IMM) 230 aus **Fig.** 2, 3 und/oder 5, um Integritätsmessungen für einen Endpunkt während der Boot-Zeit zu erzeugen. Das beispielhafte Verfahren aus **Fig.** 8 wird ausgeführt, wenn ein Endpunkt mit Implementierung des IMM 230 gestartet wird (bootet), zum Beispiel nach dem Empfangen der Konfigurationsdatei, wie das oben im Zusammenhang mit Block 10 aus **Fig.** 7 beschrieben wurde. In anderen Beispielen kann der der Neustart des Endpunkts basierend auf einer diesbezüglichen Anforderung erfolgen, die durch das SIM 228 bereitgestellt wird. In anderen Beispielen kann der Neustart des Endpunkts manuell durchgeführt werden. In Block 802 testet die beispielhafte Integritätsmessungs-Selbsttesteinrichtung 504 die Integritätsmessungsfunktionen des IMM 230. Beispielsweise testet die Integritätsmessungs-Selbsttesteinrichtung 504 den Prüfsummenalgorithmus, den Hash-Algorithmus usw., die zur Erzeugung von Integritätsmessungen verwendet werden sollen, basierend auf bekannten Eingaben und bekannten Ausgaben. In Block 804 ermittelt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506, ob die Integritätsmessungsfunktionen ordnungsgemäß funktionieren (z. B. ob die Ausgaben erwartungsgemäß sind). Wenn das nicht der Fall ist, endet das beispielhafte Verfahren aus **Fig.** 8. In einigen Beispielen erzeugt die Integritätsmessungs-Steuerungseinrichtung 506 eine Fehlermeldung, die darüber informiert, dass der Test fehlgeschlagen ist, bevor das Verfahren endet. Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 ermittelt, dass die Integritätsmessungsfunktionen ordnungsgemäß funktionieren (Block 804), wird die Steuerung mit Block 806 fortgesetzt.

[0060] In Block 806 löscht die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 alle der Integritätsmessungsregister 512, 514, 516. In Block 808 erzeugt der beispielhafte Integritätsmessungs-

generator 502 eine Integritätsmessung für einen auf den Endpunkt geladenen Software-Bestandteil (und/oder Firmware-Bestandteil). Beispielsweise berechnet der Integritätsmessungsgenerator 502 eine kryptografische Prüfsumme für den Software-Bestandteil. In Block 810 fügt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 die Integritätsmessung zum beispielhaften Software-Messungsregister 512 hinzu. In Block 812 ermittelt die beispielhafte Integritätsmessungs-Steuerungseinrichtung, ob es einen weiteren Software-Bestandteil gibt, der während des Boot-Prozesses geladen werden muss. Wenn das der Fall ist, kehrt die Steuerung zu Block 808 zurück, wo eine Integritätsmessung für diesen Software-Bestandteil erzeugt und dann zum Software-Messungsregister 512 hinzugefügt wird (Block 810). In einigen Beispielen werden die Integritätsmessungen, die mit nachfolgenden Software-Bestandteilen verknüpft sind, an die Integritätsmessungen angehängt, die mit zuvor erzeugten Integritätsmessungen für andere Software-Bestandteile an einer niedrigeren Position im Software-Stack (z. B. während der Boot-Zeit vorher geladen) verknüpft sind. In einigen Beispielen kann, wenn die angehängte Zeichenfolge von Werten zu lang für das Software-Messungsregister 512 wird, der beispielhafte Integritätsmessungsgenerator 502 einen Hashing-Algorithmus an den kombinierten Werten durchführen, um diese auf eine besser verwaltbare Menge zu reduzieren. Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung ermittelt, dass es keine weitere Software gibt, die während des Boot-Prozesses geladen werden muss (Block 812), wird die Steuerung mit Block 814 fortgesetzt. Die kombinierten Integritätsmessungen für alle Software-Bestandteile (z. B. nach einem Hashing) entsprechen der endgültigen Software-Integritätsmessung für den Endpunkt, die anschließend auf Anforderung an das SIM 228 gesendet werden kann.

[0061] In Block 814 erzeugt der beispielhafte Integritätsmessungsgenerator 502 eine Integritätsmessung für dem Endpunkt zugewiesene Konfigurationsdaten. Beispielsweise berechnet der Integritätsmessungsgenerator 502 eine kryptografische Prüfsumme für die in dem Endpunkt gespeicherte Konfigurationsdatei oder einen Abschnitt davon. In Block 816 fügt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 die Integritätsmessung zum beispielhaften Konfigurationsmessungsregister 514 hinzu. In Block 818 ermittelt die Integritätsmessungs-Steuerungseinrichtung, ob es weitere Konfigurationsdaten gibt. Wenn das der Fall ist, kehrt die Steuerung zu Block 814 zurück, wo eine Integritätsmessung für diese Konfigurationsdaten erzeugt und dann zum Konfigurationsmessungsregister 514 hinzugefügt wird (Block 816). In einigen Beispielen werden die Integritätsmessungen, die mit nachfolgend analysierten Konfigurationsdaten verknüpft sind, zum Register 514 hinzugefügt, indem sie an die Integritätsmessungen angehängt werden, die mit zuvor erzeugten Integritätsmessungen für andere Konfigurationsdaten verknüpft sind. In einigen Beispielen kann, wenn die angehängte Zeichenfolge von Werten zu lang für das Konfigurationsmessungsregister 514 wird, der beispielhafte Integritätsmessungsgenerator 502 einen Hashing-Algorithmus an den kombinierten Werten durchführen, um diese auf eine besser verwaltbare Menge zu reduzieren. Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung ermittelt, dass es keine weitere Konfiguration gibt (Block 818), wird die Steuerung mit Block 820 fortgesetzt. Die kombinierten Integritätsmessungen für alle Konfigurationsdaten (z. B. nach einem Hashing) entsprechen der endgültigen Konfigurationsintegritätsmessung für den Endpunkt, die anschließend auf Anforderung an das SIM 228 gesendet werden kann.

[0062] In Block 820 erzeugt der beispielhafte Integritätsmessungsgenerator 502 eine Integritätsmessung für eine Peripherieeinrichtung des Endpunkts. Beispielsweise berechnet der Integritätsmessungsgenerator 502 eine kryptografische Prüfsumme für eine Peripherieeinrichtung, die in der entsprechenden Konfigurationsdatei für den Endpunkt ausgewiesen ist. In Block 822 fügt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 die Integritätsmessung zum beispielhaften Hardware-Messungsregister 516 hinzu. In Block 824 ermittelt die beispielhafte Integritätsmessungs-Steuerungseinrichtung, ob es eine weitere Peripherieeinrichtung gibt. Wenn das der Fall ist, kehrt die Steuerung zu Block 820 zurück, wo eine Integritätsmessung für diese Peripherieeinrichtung erzeugt und dann zum Hardware-Messungsregister 516 hinzugefügt wird (Block 816). In einigen Beispielen werden die Integritätsmessungen, die mit nachfolgenden Peripherieeinrichtungen verknüpft sind, zum Register 516 hinzugefügt, indem sie an die Integritätsmessungen angehängt werden, die mit zuvor erzeugten Integritätsmessungen für andere Peripherieeinrichtungen verknüpft sind. In einigen Beispielen kann, wenn die angehängte Zeichenfolge von Werten zu lang für das Hardware-Messungsregister 516 wird, der beispielhafte Integritätsmessungsgenerator 502 einen Hashing-Algorithmus an den kombinierten Werten durchführen, um diese auf eine besser verwaltbare Menge zu reduzieren (z. B. in eine kürzere Zeichenfolge mit Verwendung von weniger Speicher). Die kombinierten Integritätsmessungen für alle Peripherieeinrichtungen (z. B. nach einem Hashing) entsprechen der endgültigen Hardware-Integritätsmessung für den Endpunkt, die anschließend auf Anforderung an das SIM 228 gesendet werden kann. Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung ermittelt, dass es keine weiteren Peripherieeinrichtungen gibt (Block 824), endet das beispielhafte Verfahren aus **Fig.** 8.

[0063] **Fig.** 9 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Integritätsmessungsmoduls (IMM) 230 aus **Fig.** 2, 3 und/oder 5, um Integritätsmessungen für einen Endpunkt während der Laufzeit zu erzeugen. Das heißt, das beispielhafte Verfahren aus **Fig.** 9 erfolgt, nachdem ein Endpunkt konfiguriert und für ein Netzwerk zugelassen wurde und sich in Betrieb befindet. Das beispielhafte Verfahren aus **Fig.** 9 beginnt in Block 902, wo die beispielhafte Kommunikationsschnittstelle 510 eine Anforderung von einem Systemintegritätsüberwachungsmodul (z. B. dem SIM 228 aus **Fig.** 2, 3 und/oder 6) empfängt. In einigen Beispielen enthält die Anforderung einen Nonce-Wert, der bei der Erzeugung der Antwort zu verwenden ist. Weitere Details in Bezug auf das SIM 228 und Szenarien, wann eine solche Anforderung empfangen werden kann, werden im Folgenden im Zusammenhang mit **Fig.** 11-13 beschreiben. In Block 904 testet die beispielhafte Integritätsmessungs-Selbsttesteinrichtung 504 die Integritätsmessungsfunktionen des IMM 230. In Block 906 ermittelt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506, ob die Integritätsmessungsfunktionen ordnungsgemäß funktionieren (z. B. ob die Ausgaben erwartungsgemäß sind). Wenn das nicht der Fall ist, endet das beispielhafte Verfahren aus **Fig.** 9. In einigen Beispielen erzeugt die Integritätsmessungs-Steuerungseinrichtung 506 eine Fehlermeldung, die darüber informiert, dass der Test fehlgeschlagen ist, bevor das Verfahren endet. Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 ermittelt, dass die Integritätsmessungsfunktionen ordnungsgemäß funktionieren (Block 906), wird die Steuerung mit Block 908 fortgesetzt.

[0064] In Block 908 löscht die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 die Konfigurations- und Hardware-Integritätsmessungsregister 514, 516. In einigen Beispielen wird das Software-Messungsregister 512 in Verbindung mit einer Laufzeit-Integritätsmessung, wie sie in dem beispielhaften Verfahren aus **Fig.** 9 implementiert ist, nicht gelöscht, weil Software-Integritätsmessungen erzeugt werden, wenn der Endpunkt während der Boot-Zeit Software lädt. Auf diese Weise können zuvor erfasste, im Software-Messungsregister 512 gespeicherte Software-Integritätsmessungen, die während des letzten Boot-Vorgangs des Endpunkts erzeugt wurden, weiterhin für Berichtszwecke verfügbar sein.

[0065] In Block 910 erzeugt der beispielhafte Integritätsmessungsgenerator 502 eine Integritätsmessung für dem Endpunkt zugewiesene Konfigurationsdaten. In einigen Beispielen basieren die dem Endpunkt zugewiesenen Konfigurationsdaten auf dem Konfigurationsprofil und/oder Abschnitten davon, das auf dem Endpunkt gespeichert ist. In Block 912 fügt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 die Integritätsmessung zum beispielhaften Konfigurationsmessungsregister 514 hinzu. In Block 914 ermittelt die Integritätsmessungs-Steuerungseinrichtung, ob es weitere Konfigurationsdaten gibt. Wenn das der Fall ist, kehrt die Steuerung zu Block 910 zurück, wo eine Integritätsmessung für diese Konfigurationsdaten erzeugt und dann zum Konfigurationsmessungsregister 514 hinzugefügt wird (Block 912). In einigen Beispielen werden die Integritätsmessungen, die mit nachfolgend analysierten Konfigurationsdaten verknüpft sind, zum Register 514 hinzugefügt, indem sie an die Integritätsmessungen angehängt werden, die mit zuvor erzeugten Integritätsmessungen für andere Konfigurationsdaten verknüpft sind. In einigen Beispielen kann, wenn die angehängten Werte zu lang für das Konfigurationsmessungsregister 514 werden, der beispielhafte Integritätsmessungsgenerator 502 einen Hashing-Algorithmus an den kombinierten Werten durchführen, um diese auf eine besser verwaltbare Menge zu reduzieren. Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung ermittelt, dass es keine weiteren Konfigurationsdaten gibt (Block 914), wird die Steuerung mit Block 916 fortgesetzt. Die kombinierten Integritätsmessungen für alle dem Endpunkt zugewiesenen Konfigurationsdaten (z. B. nach einem Hashing) entsprechen der endgültigen Konfigurationsintegritätsmessung für den Endpunkt.

[0066] In Block 916 erzeugt der beispielhafte Integritätsmessungsgenerator 502 eine Integritätsmessung für eine Peripherieeinrichtung des Endpunkts. In Block 918 fügt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 die Integritätsmessung zum beispielhaften Hardware-Messungsregister 516 hinzu. In Block 920 ermittelt die Integritätsmessungs-Steuerungseinrichtung, ob es eine weitere Peripherieeinrichtung gibt. Wenn das der Fall ist, kehrt die Steuerung zu Block 916 zurück, wo eine Integritätsmessung für diese Peripherieeinrichtung erzeugt und dann zum Hardware-Messungsregister 516 hinzugefügt wird (Block 918). In einigen Beispielen werden die Integritätsmessungen, die mit nachfolgenden Peripherieeinrichtungen verknüpft sind, zum Hardware-Messungsregister 516 hinzugefügt, indem sie an die Integritätsmessungen angehängt werden, die mit zuvor erzeugten Integritätsmessungen für andere Peripherieeinrichtungen verknüpft sind. In einigen Beispielen kann, wenn die angehängten Werte zu lang für das Hardware-Messungsregister 516 werden, der beispielhafte Integritätsmessungsgenerator 502 einen Hashing-Algorithmus an den kombinierten Werten durchführen, um diese auf eine besser verwaltbare Menge zu reduzieren. Die kombinierten Integritätsmessungen für alle Peripherieeinrichtungen (z. B. nach einem Hashing) entsprechen der endgültigen Konfigurationsintegritätsmessung für den Endpunkt.

[0067] Wenn die beispielhafte Integritätsmessungs-Steuerungseinrichtung ermittelt, dass es keine weiteren Peripherieeinrichtungen gibt (Block 920), wird die Steuerung mit Block 922 fortgesetzt, wo die beispielhafte Kommunikationsschnittstelle 510 einen Bericht der Integritätsmessungen sendet. In einigen Beispielen enthalten die berichteten Integritätsmessungen die im Software-Messungsregister 512 gespeicherten Software-Integritätsmessungen (zuvor während des letzten Boot-Vorgangs des Endpunkts erzeugt), die im Konfigurationsmessungsregister 514 gespeicherten Konfigurationsintegritätsmessungen (in Block 912 erzeugt) und die im Hardware-Messungsregister 516 gespeicherten Hardware-Integritätsmessungen (in Block 918 erzeugt). In einigen Beispielen wird der Bericht der Integritätsmessungen zusammen mit dem Nonce-Wert bereitgestellt, der mit der Anforderung für die Messungen bereitgestellt wurde. Nach dem Berichten der Integritätsmessungen endet das beispielhafte Verfahren aus **Fig.** 9.

[0068] **Fig.** 10 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung der beispielhaften Integritätsmessungsmodule (IMM) 230 aus **Fig.** 2, 3 und/oder 5, um die Verwendung von Autorisierungsinformationen an einen Endpunkt auf Umstände einzuschränken, wenn sich der Endpunkt in einem vertrauenswürdigen Status befindet. Wie oben beschreiben wurde, wird die Kommunikation über Autorisierungsinformationen ermöglicht, die an einen Endpunkt bereitgestellt werden, aus denen der Endpunkt geheime und/oder öffentliche Werte oder Schlüssel erzeugt. In einigen Beispielen speichert der Endpunkt die Autorisierungsinformationen in einer verschlüsselten Form, auf die nur zugegriffen werden kann (entschlüsselt werden können), wenn sich der Endpunkt im selben Status befindet wie zum Zeitpunkt der Verschlüsselung der Autorisierungsinformationen. In einigen Beispielen werden die Autorisierungsinformationen dem Endpunkt bereitgestellt, nachdem die Integrität des Endpunkts verifiziert wurde, sodass sich der Endpunkt in einem vertrauenswürdigen Status befindet, wenn die Autorisierungsinformationen verschlüsselt werden. Damit ist, in einigen Bespielen, wenn ein Endpunkt kompromittiert ist und sich damit nicht mehr in einem vertrauenswürdigen Status befindet (z. B. Integritätsmessungen würden nicht mehr mit Referenzwerten übereinstimmen), der Endpunkt nicht in der Lage, die Autorisierungsinformationen zu verwenden, weil der Endpunkt die Autorisierungsinformationen nicht entschlüsseln kann. Auf diese Weise kann die Kommunikation von einem potenziell kompromittierten Endpunkt verhindert werden, ohne dass der Endpunkt speziell durch ein SIM 228 getestet werden muss, wie das im Folgenden ausführlicher beschrieben wird.

[0069] Das beispielhafte Verfahren aus **Fig.** 10 beginnt, wenn ein Endpunkt mit Implementierung des IMM 230 gestartet wird (bootet). Im dargestellten Beispiel ist das ein Boot-Vorgang, nachdem das IMM 230 bereits konfiguriert wurde und Autorisierungsinformationen empfangen hat, die in einer verschlüsselten Form in der Autorisierungsinformationsdatenbank 518 gespeichert wurden. Das beispielhafte Verfahren aus **Fig.** 10 beginnt mit Block 1002, wo der beispielhafte Integritätsmessungsgenerator 502 Boot-Zeit-Integritätsmessungen erfasst (z. B. durch Implementierung des beispielhaften Verfahrens aus **Fig.** 8). In Block 1004 ruft der beispielhafte Kommunikationsauthentifikator 508 verschlüsselte Autorisierungsinformationen aus der Autorisierungsinformationsdatenbank 518 ab. In Block 1006 versucht der beispielhafte Kommunikationsauthentifikator 508, die Autorisierungsinformationen auf der Basis der Boot-Zeit-Integritätsmessungen zu dechiffrieren. Das heißt, dass, in einigen Beispielen, der Kommunikationsauthentifikator 508 den durch die Integritätsmessungen angezeigten Status des Endpunkts verwendet, um die verschlüsselten Autorisierungsinformationen zu entschlüsseln.

[0070] In Block 1008 ermittelt der beispielhafte Kommunikationsauthentifikator 508, ob die Autorisierungsinformationen erfolgreich entschlüsselt wurden. Wenn das der Fall ist, zeigt dies an, dass der Status des Endpunkts dem erwarteten Status der Integrität entspricht (z. B. einem vertrauenswürdigen Status), und der Endpunkt kann auf die Autorisierungsinformationen zugreifen und diese damit verwenden, um zu kommunizieren. Dementsprechend verwendet in Block 1010 die beispielhafte Kommunikationsschnittstelle 510 die dechiffrierten Autorisierungsinformationen für die Kommunikation innerhalb des Steuerungssystems. Danach endet das beispielhafte Verfahren aus **Fig.** 10.

[0071] Wenn der beispielhafte Kommunikationsauthentifikator 508 ermittelt, dass die Autorisierungsinformationen nicht erfolgreich dechiffriert wurden (Block 1008), wird die Steuerung mit Block 1012 fortgesetzt. Ein erfolgloser Versuch zur Dechiffrierung der verschlüsselten Autorisierungsinformationen zeigt an, dass etwas beim Status des Endpunkts (wie in den Integritätsmessungen angezeigt) vom gewünschten Status (z. B. Integrität) des Endpunkts abweicht. In Block 1012 ermittelt die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506, ob die Konfigurationsdatei verfügbar ist. Wenn das der Fall ist, wird die Steuerung mit Block 1014 fortgesetzt, wo die beispielhafte Kommunikationsschnittstelle 510 eine Anforderung für die Zulassung zum Steuerungsnetzwerk sendet. In einigen Beispielen kann diese Anforderung das im Folgenden beschriebene beispielhafte Verfahren aus **Fig.** 12 einleiten. Danach endet das beispielhafte Verfahren aus **Fig.** 10. Wenn dagegen wiederum in Block 1012 die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 ermittelt, dass die Konfigurationsdatei

nicht verfügbar ist, wird die Steuerung mit Block 1016 fortgesetzt, bevor das beispielhafte Verfahren aus **Fig.** 10 endet. In Block 1016 sendet die beispielhafte Kommunikationsschnittstelle 510 eine Anforderung für die Registrierung im Steuerungsnetzwerk. In einigen Beispielen entspricht diese Anforderung dem Beginn des im Folgenden beschriebenen beispielhaften Verfahrens aus **Fig.** 11.

[0072] **Fig.** 11 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls 228 aus **Fig.** 2, 3 und/oder 6, um die Integrität eines Endpunkts zu überprüfen, der sich erstmalig in einem zugehörigen Steuerungsnetzwerk registriert. Das beispielhafte Verfahren beginnt in Block 1102, wo die beispielhafte Kommunikationsschnittstelle 602 eine Anforderung zur Registrierung von einem Endpunkt empfängt. Die Registrierung des Endpunkts bezieht sich auf die Verbindung und Identifizierung des Endpunkts als ein Knoten oder Endpunkt in dem Steuerungsnetzwerk entsprechend dem Steuerungssystem, in dem der Endpunkt implementiert werden soll, der zuvor noch nicht verbunden war oder nicht über gültige Konfigurationsdaten verfügt (z. B. die Anforderung in Block 1016 aus **Fig.** 10). In Block 1104 sendet die beispielhafte Kommunikationsschnittstelle 602 eine Nonce (z. B. erzeugt durch die beispielhafte Autorisierungssteuerungseinrichtung 606) an den Endpunkt zusammen mit einer Anforderung für Software-Integritätsmessungen für den Endpunkt. In einigen Beispielen dient die Nonce dazu, einen „lügenden Endpunkt“ zu erkennen (z. B. einen Endpunkt, der einfach nur zuvor ermittelte Werte wiederholt). Im dargestellten Beispiel wird davon ausgegangen, dass der Endpunkt eine neu implementierte Einrichtung ist, die noch nicht konfiguriert wurde (oder eine Einrichtung, die über keine gültigen Konfigurationsdaten mehr verfügt), sodass die entsprechenden Konfigurations- und/oder Hardware-Integritätsmessungen keine gültigen Ergebnisse erzeugen würden, die mit den entsprechenden Referenzwerten übereinstimmen. Somit werden in dem dargestellten Beispiel Software-Integritätsmessungen angefordert, während andere Typen von Integritätsmessungen (z. B. Hardware- und/oder Konfigurationsintegritätsmessungen) nicht angefordert werden. Allerdings können, in einigen Beispielen, die Hardware- und/oder Konfigurationsintegritätsmessungen trotzdem zusammen mit den Software-Integritätsmessungen in Block 1104 angefordert werden.

[0073] In Block 1106 empfängt die beispielhafte Kommunikationsschnittstelle 602 einen Bericht der Software-Integritätsmessungen. In einigen Beispielen entsprechen die berichteten Software-Integritätsmessungen beliebigen Werten, die im Software-Messungsregister 512 gespeichert sind, deren Erzeugung zuvor erfolgte, als der Endpunkt erstmalig gestartet wurde, wie das oben im Zusammenhang mit **Fig.** 8 beschrieben wurde. In Block 1108 speichert die beispielhafte Integritätsmessungsdatenbank 618 die berichteten Software-Integritätsmessungen. In einigen Beispielen werden die Integritätsmessungen zum nachfolgenden Vergleich mit Referenzwerten und/oder zur Integritätsverifizierung des gesamten Steuerungssystems (als integriertes Ganzes) gegenüber einem Beaufsichtigungssystem verwendet, wie das im Folgenden im Zusammenhang mit **Fig.** 16 ausführlicher beschrieben wird.

[0074] In Block 1110 vergleicht der beispielhafte Integritätsmessungskomparator 604 die Software-Integritätsmessungen mit Software-Referenzwerten. In einigen Beispielen wurden die Software-Referenzwerte zuvor durch einen Benutzer über die Benutzerschnittstelle 614 basierend auf den Daten eingegeben, die durch den Hersteller der Endpunkteinrichtung bereitgestellt wurden. Auf dieses Weise können die Referenzwerte unabhängig von direktem Feedback von dem Endpunkt festgelegt werden, wodurch die Möglichkeit entfällt, dass die Endpunkteinrichtung manipuliert wurde, nachdem sie den Besitzbereich des Herstellers verlassen hatte und bevor der Endpunkt mit dem Netzwerk verbunden wird (z. B. während des Versands). In einigen Beispielen können die Hardware- und/oder Software-Konfigurationsreferenzwerte auch durch einen Ingenieur oder einen anderen Mitarbeiter eingegeben werden. In anderen Beispielen werden die Software-Referenzwerte basierend auf den erstmaligen Software-Integritätsmessungen gewonnen, die in Block 1106 empfangen werden, wie das im Folgenden ausführlicher beschrieben wird. In gleicher Weise können, in einigen Beispielen, die Ausgangsdaten zum Festlegen der Hardware- und/oder Software-Konfigurationsreferenzwerte aus einer erstmaligen Berechnung der entsprechenden Integritätsmessungen gewonnen werden.

[0075] In Block 1112 ermittelt der beispielhafte Integritätsmessungskomparator 604, ob die berichteten Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn die Referenzwerte zuvor manuell durch einen Benutzer eingegeben wurden und der Endpunkt seit dem Verlassen des Herstellers nicht verändert wurde, sollten die Software-Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn das der Fall ist, wird die Steuerung mit Block 1114 fortgesetzt, wo die beispielhafte Autorisierungssteuerungseinrichtung 606 den Endpunkt für den Kommunikationszugriff zur Konfiguration autorisiert. In einigen Beispielen ermöglicht der Kommunikationszugriff zur Konfiguration dem Endpunkt das Empfangen von Konfigurationsdaten, die zum Konfigurieren des Endpunkts verwendet werden sollen (z. B. basierend auf der im Zusammenhang mit **Fig.** 7 oben beschriebenen Konfigurationsdatei).

[0076] Wenn die Integritätsmessungen nicht mit den Referenzwerten übereinstimmen (Block 1112), wird die Steuerung mit Block 1116 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 einen Administrator auffordert, die Diskrepanz zu überschreiben. In einigen Beispielen stimmen die Integritätsmessungen eventuell nicht mit den Referenzwerten überein, wenn ein Endpunkt erstmalig implementiert wird (z. B. eine neu erworbene und/oder installierte Einrichtung) und die Referenzwerte nicht zuvor manuell über die Benutzerschnittstelle 614 eingegeben wurden. Dementsprechend ermöglicht eine in Block 1116 bereitgestellte Aufforderung einem Administrator, die anfänglich berichteten Integritätsmessungen als die neuen Referenzwerte zu akzeptieren, wenn sie den gewünschten Werten entsprechen. Somit ermittelt in Block 1118 die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob die Überschreibung bestätigt ist. Wenn das der Fall ist, wird die Steuerung mit Block 1120 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die berichteten Integritätsmessungen als die neuen Referenzwerte zuweist. Auf diese Weise können die Software-Referenzwerte anfänglich basierend auf den Integritätsmessungen des Endpunkts festgelegt werden, ohne dass eine manuelle Eingabe der Werte wie oben beschrieben erforderlich ist. Weil in Block 1120 die berichteten Integritätsmessungen als die Referenzwerte zugewiesen werden, stimmen in solchen Beispielen die Integritätsmessungen notwendigerweise mit den Referenzwerten überein, sodass die Steuerung mit Block 1114 fortgesetzt wird, um den Endpunkt für den Kommunikationszugriff zur Konfiguration zu autorisieren.

[0077] Sobald der Endpunkt für den Kommunikationszugriff zur Konfiguration zu autorisiert wurde (Block 1114), kann die Konfigurationsdatei von dem Konfigurationsmodul 226 zu dem Endpunkt gesendet werden, wie das oben im Zusammenhang mit **Fig.** 7 beschrieben wurde. Danach kann ein Neustart des Endpunkts durchgeführt werden, um den Konfigurationsprozess abzuschließen. Auf diese Weise kann der Endpunkt gültige Konfigurationsdaten erwerben, sodass die Software-, die Hardware- und die Konfigurationsintegritätsmessung jeweils den ordnungsgemäßen Status des Endpunkts zur Verifizierung der Integrität des Endpunkts widerspiegeln. In Block 1122 empfängt die beispielhafte Kommunikationsschnittstelle 602 von dem Endpunkt nach einem Konfigurationsneustart eine Anforderung zur Zulassung. Die Zulassung des Endpunkts bezieht sich darauf, dass der Endpunkt zugelassen wird für die Kommunikation (z. B. die Kommunikation ermöglicht wird) mit anderen Endpunkten in diesem Steuerungsnetzwerk, das dem Steuerungssystem zugeordnet ist, in dem der Endpunkt implementiert werden soll und in dem der Endpunkt bereits registriert ist (z. B. in Block 1114 als Reaktion auf die Anforderung zur Registrierung in Block 1102). In Block 1124 verifiziert die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die Integrität des vollständig konfigurierten Endpunkts. Ein beispielhaftes Verfahren zur Verifizierung der Integrität des vollständig konfigurierten Endpunkts wird im Folgenden in **Fig.** 12 ausführlicher beschrieben. In Block 1126 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob es einen weiteren Endpunkt gibt, der für das Steuerungsnetzwerk des Steuerungssystems zugelassen werden soll. Wenn das der Fall ist, kehrt die Steuerung zu Block 1102 zurück. Andernfalls endet das beispielhafte Verfahren aus **Fig.** 11.

[0078] Wenn wiederum in Block 1118 die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass die Überschreibung nicht bestätigt ist (z. B. wenn der Administrator die vom Endpunkt berichtete Software-Integritätsmessung nicht als den neuen Referenzwert verwenden möchte), wird die Steuerung mit Block 1128 fortgesetzt, wo die beispielhafte Autorisierungssteuerungseinrichtung 606 den Endpunkt auf den Kommunikationszugriff für Firmware-Updates einschränkt. In einigen Beispielen schränkt der Kommunikationszugriff für Firmware-Updates den Endpunkt auf den Empfang von Firmware-Updates ein, erlaubt aber keinerlei sonstige Kommunikation mit dem Endpunkt. Nach dem Einschränken des Kommunikationszugriffs für den Endpunkt (Block 1128) wird die Steuerung mit Block 1126 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung ermittelt, ob das beispielhafte Verfahren beendet oder für einen weiteren Endpunkt wiederholt wird.

[0079] **Fig.** 12 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls 228 aus **Fig.** 2, 3 und/oder 6, um die Integrität eines Endpunkts zu überprüfen, der versucht, Kommunikationszugriff in einem Steuerungsnetzwerk eines Steuerungssystems zu erhalten. Insbesondere entspricht das beispielhafte Verfahren aus **Fig.** 12 dem Block 1124 aus **Fig.** 11. Das beispielhafte Verfahren aus **Fig.** 12 beginnt in Block 1202, wo die beispielhafte Kommunikationsschnittstelle 602 eine Nonce (z. B. erzeugt durch die beispielhafte Autorisierungssteuerungseinrichtung 606) an den Endpunkt zusammen mit einer Anforderung für Integritätsmessungen für den Endpunkt sendet. Im Gegensatz zum Block 1104 aus **Fig.** 11, wo nur Software-Integritätsmessungen angefordert wurden, kann die beispielhafte Kommunikationsschnittstelle 602 Werte jeweils für die Software-, die Hardware- und die Konfigurationsintegritätsmessungen anfordern, weil der Endpunkt an dieser Stelle vollständig konfiguriert ist und die Referenzwertdatenbank 616

über alle relevanten Referenzwerte zum Vergleich mit solchen Integritätsmessungen verfügt. In einigen Beispielen wird zuerst die Verifizierung der Software (**Fig.** 11) durchgeführt, um sicherzustellen, dass der Endpunkt bei seiner Bereitstellung durch den Hersteller erwartungsgemäß funktioniert, bevor verifiziert wird, dass der Endpunkt für die konkret Anwendung, für die der Endpunkt verwendet wird, ordnungsgemäß konfiguriert wurde (**Fig.** 12). In einigen Beispielen können die beispielhaften Verfahren aus **Fig.** 11 und 12 kombiniert werden, sodass die Software-, Hardware- und Konfigurationsintegrität des Endpunkts gleichzeitig verifiziert werden.

[0080] In Block 1204 empfängt die beispielhafte Kommunikationsschnittstelle 602 einen Bericht der Integritätsmessungen. In einigen Beispielen basieren die Integritätsmessungen auf Berechnungen, die während der Laufzeit erfolgt sind. In einigen solcher Beispiele entsprechen die berichteten Software-Integritätsmessungen zuvor im Software-Messungsregister 512 gespeicherten Werten, weil kein Neustart des Endpunkts durchgeführt wurde, um neue Messungen zu erhalten. Das heißt, dass die Software-Integritätsmessungen den Werten entsprechen können, die zuvor erzeugt wurden, als der Endpunkt letztmalig neu gestartet wurde (z. B. unmittelbar vor Block 1122 in **Fig.** 11). Im Gegensatz dazu werden, in einigen Beispielen, die Hardware- und Konfigurationsintegritätsmessungen während der Laufzeit neu erzeugt. In Block 1206 speichert die beispielhafte Integritätsmessungsdatenbank 618 die berichteten Integritätsmessungen.

[0081] In Block 1208 vergleicht der beispielhafte Integritätsmessungskomparator 604 die berichteten Integritätsmessungen mit Referenzwerten. In einigen Beispielen entsprechen die Referenzwerte den Werten, die in Block 1102 empfangen wurden und/oder in Block 1120 von **Fig.** 11 zugewiesen wurden. In Block 1210 ermittelt der beispielhafte Integritätsmessungskomparator 604, ob die berichteten Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn das nicht der Fall ist, wird die Steuerung mit Block 1212 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 einen Administrator auffordert, die Diskrepanz zu überschreiben. In einigen Beispielen ermöglicht die Aufforderung einem Administrator, die berichteten Integritätsmessungen als die neuen Referenzwerte zu akzeptieren. In Block 1214 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob die Überschreibung bestätigt ist. Wenn das der Fall ist, wird die Steuerung mit Block 1216 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die berichteten Integritätsmessungen als die neuen Referenzwerte zuweist, woraufhin die Steuerung mit Block 1218 fortgesetzt wird.

[0082] In Block 1218 autorisiert die beispielhafte Autorisierungssteuerungseinrichtung 606 den Endpunkt zum vollständigen Kommunikationszugriff, indem sie dem Endpunkt Autorisierungsinformationen bereitstellt. In einigen Beispielen ermöglicht der vollständige Kommunikationszugriff dem Endpunkt das vollständige Kommunizieren mit anderen Endpunkten innerhalb des Steuerungsnetzwerks, in dem der Endpunkt implementiert wird. In einigen Beispielen wird der vollständige Kommunikationszugriff ermöglicht, indem dem Endpunkt Autorisierungsinformationen bereitgestellt werden, die von dem Endpunkt verwendet werden, um geheime und/oder öffentliche Werte zu erzeugen. Wenn wiederum in Block 1210 der beispielhafte Integritätsmessungskomparator 604 ermittelt, dass die berichteten Integritätsmessungen nicht mit den Referenzwerten übereinstimmen, wird die Steuerung direkt mit Block 1218 fortgesetzt. Sobald der Endpunkt zum vollständigen Kommunikationszugriff autorisiert wurde (Block 1218), endet das beispielhafte Verfahren aus **Fig.** 12 und kehrt zum Abschließen des beispielhaften Verfahrens aus **Fig.** 11 zurück.

[0083] Wenn wiederum in Block 1214 die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass die Überschreibung nicht bestätigt ist (z. B. wenn der Administrator die berichtete Integritätsmessung nicht als den neuen Referenzwert verwenden möchte), wird die Steuerung mit Block 1220 fortgesetzt, wo der beispielhafte Integritätsmessungskomparator 604 ermittelt, ob die berichteten Software-Integritätsmessungen mit den Software-Referenzwerten übereinstimmen. Wenn das nicht der Fall ist, ist die Software auf dem Endpunkt verdächtig. Dementsprechend wird die Steuerung mit Block 1222 fortgesetzt, wo die beispielhafte Autorisierungssteuerungseinrichtung 606 den Endpunkt auf den Kommunikationszugriff für Firmware-Updates einschränkt. Nach dem Einschränken des Kommunikationszugriffs für den Endpunkt (Block 1222) endet das beispielhafte Verfahren aus **Fig.** 12 und kehrt zum Abschließen des beispielhaften Verfahrens aus **Fig.** 11 zurück. Wenn wiederum in Block 1220 der beispielhafte Integritätsmessungskomparator 604 ermittelt, dass die berichteten Software-Integritätsmessungen nicht mit den Software-Referenzwerten übereinstimmen, wird die Steuerung mit Block 1224 fortgesetzt. In Block 1224 schränkt die beispielhafte Autorisierungssteuerungseinrichtung 606 den Endpunkt auf den Kommunikationszugriff zur Konfiguration ein. Danach endet das beispielhafte Verfahren aus **Fig.** 12 und kehrt zum Abschließen des beispielhaften Verfahrens aus **Fig.** 11 zurück.

[0084] **Fig.** 13 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls 228 aus **Fig.** 2, 3 und/oder 6, um die

Integrität eines Endpunkts in einem Steuerungsnetzwerk eines Steuerungssystems während der Laufzeit zu überprüfen. Das heißt, während **Fig.** 11 und 12 ein beispielhaftes Verfahren repräsentieren, um die Integrität eines Endpunkts zu verifizieren, der in einem Steuerungsnetzwerk erstmalig eingerichtet und konfiguriert wird, repräsentiert **Fig.** 13 die Verifizierung der Integrität des Endpunkts zu einem späteren Zeitpunkt, während sich der Endpunkt im Betrieb und/oder im Dienst befindet. Auf diese Weise können potenzielle Sicherheitsverstöße erkannt werden, die auftreten, nachdem der Endpunkt erworben, installiert und zur Verwendung konfiguriert wurde, und es kann entsprechend darauf reagiert werden.

[0085] Das beispielhafte Verfahren aus **Fig.** 13 beginnt in Block 1302, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, ob es an der Zeit ist, die Laufzeitintegrität eines Endpunkts zu verifizieren. Viele Computereinrichtungen (Endpunkte) in einem Steuerungssystem können über längere Zeiträume in Betrieb bleiben (z. B. Monate oder Jahre), ohne dass sie ausgeschaltet oder neu gestartet werden. Damit werden, obwohl die Integrität eines Endpunkts während der Boot-Zeit nützlich ist, um den Status des Software-Stacks (beim Boot-Vorgang geladen) zu bestätigen, solche Tests oftmals nicht ausreichend regelmäßig durchgeführt, um potenzielle Änderungen an dem Endpunkt zu erkennen. Dementsprechend kann, in einigen Beispielen, der Benutzer die Laufzeitintegrität in einem periodischen oder aperiodischen Zeitplan definieren, der regelmäßiger und/oder häufiger ist. In einigen Beispielen werden die Laufzeitintegritätsmessungen für jeden Endpunkt in einem Netzwerk einmal täglich durchgeführt. Allerdings können die Laufzeitintegritätstests in jedem geeigneten Zeitplan implementiert werden. Darüber hinaus können, in einigen Beispielen, bestimmte Endpunkte häufiger als andere getestet werden, was sich zum Beispiel danach richtet, wie kritisch der Endpunkt für das Steuerungssystem ist, in dem er betrieben wird (z. B. stündlich für kritische Endpunkte). Zusätzlich oder alternativ werden, in einigen Beispielen, nur bestimmte Typen von Integritätsmessungen (Software, Hardware oder Konfiguration) zu einem bestimmten Zeitpunkt angefordert. In einigen Beispielen kann ein getrennter Zeitplan für die Überwachung jeweils der Software-, Hardware- und Konfigurationsintegritätsmessungen angewendet werden. Wenn die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass es nicht an der Zeit ist, die Laufzeitintegrität eines Endpunkts zu verifizieren (Block 1302), wartet die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, bis es an der Zeit ist.

[0086] Sobald es an der Zeit ist, die Laufzeitintegrität des Endpunkts zu verifizieren, wird die Steuerung mit Block 1304 fortgesetzt, wo die beispielhafte Kommunikationsschnittstelle 602 eine Nonce (z. B. erzeugt durch die beispielhafte Autorisierungssteuerungseinrichtung 606) an den Endpunkt zusammen mit einer Anforderung für Integritätsmessungen für den Endpunkt sendet. In Block 1306 empfängt die beispielhafte Kommunikationsschnittstelle 602 einen Bericht der Integritätsmessungen. In einigen Beispielen basieren die Integritätsmessungen auf Berechnungen, die während der Laufzeit erfolgt sind. Allerdings basieren die Software-Integritätsmessungen, in einigen Beispielen, auf Integritätsmessungen, die erzeugt wurden, als der Endpunkt letztmalig einen Neustart ausführte, sodass während der Laufzeit keine neuen Software-Integritätsmessungen erstellt werden. In solchen Beispielen entsprechen die Software-Integritätsmessungen vielmehr den Werten, die zuvor erzeugt wurden, als der Endpunkt letztmalig neu gestartet wurde. Im Gegensatz dazu werden, in einigen Beispielen, die Hardware- und Konfigurationsintegritätsmessungen während der Laufzeit neu erzeugt. In Block 1308 speichert die beispielhafte Integritätsmessungsdatenbank 618 die berichteten Integritätsmessungen.

[0087] In Block 1310 vergleicht der beispielhafte Integritätsmessungskomparator 604 die berichteten Integritätsmessungen mit Referenzwerten. In Block 1312 ermittelt der beispielhafte Integritätsmessungskomparator 604, ob die berichteten Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn das nicht der Fall ist, wird die Steuerung mit Block 1314 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 basierend auf den fehlgeschlagenen Integritätsmessungen eine Reaktion implementiert. In einigen Beispielen hängt die konkrete Reaktion davon ab, welcher Typ von Integritätsmessungen (z. B. Software-, Hardware- und/oder Konfiguration) nicht mit den Referenzwerten übereinstimmt. In einigen Beispielen hängt die Art und Weise der Reaktion von der Art und Weise des Endpunkts und/oder seiner Konfiguration und Beziehung zu anderen Endpunkten in dem entsprechenden Netzwerk ab. In einigen Beispielen umfasst die Reaktion das Erzeugen einer Warnung (Block 1316), das Markieren künftiger Daten von dem Endpunkt als verdächtig (Block 1318), das Einschränken des Endpunkts auf den Kommunikationszugriff für Firmware-Updates und/oder den Kommunikationszugriff zur Konfiguration (Block 1320) und/oder das Umschalten zu einem redundanten Endpunkt (Block 1322). In einigen Beispielen umfasst die Reaktion eine Kombination aus mehr als einem der Blöcke 1316, 1318, 1320, 1322. In einigen Beispielen wird die implementierte Reaktion durch einen Benutzer definiert, wenn der Endpunkt konfiguriert wird. Nachdem die entsprechende Reaktion implementiert wurde (Block 1314), wird die Steuerung mit Block 1324 fortgesetzt. Wenn wiederum in Block 1312 der beispielhafte Integritätsmessungskomparator 604 ermittelt, dass die Integritäts-

messungen mit den Referenzwerten übereinstimmen, ist die Integrität des Endpunkts verifiziert, und keine weiteren Aktionen sind erforderlich. In solchen Beispielen wird die Steuerung direkt mit Block 1324 fortgesetzt.

[0088] In Block 1324 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob es einen weiteren Endpunkt gibt, der verifiziert werden muss. Wenn das der Fall ist, kehrt die Steuerung zu Block 1302 zurück. Andernfalls wird die Steuerung mit Block 1326 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, ob die Steuerung fortgesetzt werden soll. Wenn das der Fall ist, kehrt die Steuerung zu Block 1302 zurück. Andernfalls endet das beispielhafte Verfahren aus **Fig.** 13.

[0089] Die oben beschriebenen **Fig.** 11-13 entsprechen der Implementierung des SIM 228 innerhalb einer primären Arbeitsstation eines Steuerungssystems (z. B. des Steuerungssystems 200 aus **Fig.** 2), die in der Steuerungssystemebene 106 der Hierarchie 100 aus **Fig.** 1 angesiedelt ist. Damit beziehen sich **Fig.** 11-13 auf die konfigurations- und integritätsbasierte Autorisierung des Kommunikationszugriffs für Prozesssteuerungseinrichtungen (z. B. Steuerungseinrichtungen, E/A-Einrichtungen, Feldeinrichtungen usw.) auf der Einrichtungsebene 104 der Hierarchie 100 in einem Steuerungsnetzwerk, das zur Steuerungssystemebene 106 der Hierarchie 100 gehört. Zusätzlich oder alternativ beziehen sich **Fig.** 11-13 auf die konfigurations- und integritätsbasierte Autorisierung der Kommunikation zwischen verschiedenen Endpunkten (z. B. Arbeitsstationen, Servern usw.) auf der Steuerungssystemebene 106 über einem Steuerungsnetzwerk, das zur Steuerungssystemebene 106 gehört. Damit wird, in einigen Beispielen, das mit **Fig.** 11-13 verbundene SIM 228 in einer primären Arbeitsstation implementiert, die für das Steuerungsnetzwerk ausgewiesen ist, für das die Kommunikation autorisiert wird (z. B. die primäre Arbeitsstation 220 aus **Fig.** 2). Des Weiteren ist, in solchen Beispielen, das IMM 230 in Endpunkten implementiert, die konkreten Einrichtungen entsprechen, die kommunikationsfähig mit dem Steuerungsnetzwerk gekoppelt sind.

[0090] Im Gegensatz dazu entsprechen die im Folgenden beschriebenen **Fig.** 14-15 der Implementierung des SIM 228 innerhalb einer primären Arbeitsstation eines Beaufsichtigungssystems (z. B. des Beaufsichtigungssystems 300 aus **Fig.** 3), die in der Beaufsichtigungssystemebene 108 der Hierarchie 100 aus **Fig.** 1 angesiedelt ist. Damit beziehen sich **Fig.** 14-15 auf die integritätsbasierte Autorisierung des Kommunikationszugriffs für Endpunkte in einem Beaufsichtigungssystem (z. B. dem Beaufsichtigungssystem 300 aus **Fig.** 3), die zur Beaufsichtigungssystemebene 108 der Hierarchie 100 aus **Fig.** 1 gehören, und/oder zwischen einem untergeordneten System (z. B. dem Steuerungssystemen 200, 302 aus **Fig.** 3) auf der Steuerungssystemebene 106 und anderen Endpunkten auf der Beaufsichtigungssystemebene 108. Damit wird, in einigen Beispielen, das in Verbindung mit **Fig.** 14-15 referenzierte SIM 228 in einer primären Arbeitsstation implementiert, die für das Beaufsichtigungsnetzwerk ausgewiesen ist, für das die Kommunikation autorisiert wird (z. B. die primäre Arbeitsstation 306 aus **Fig.** 3). Des Weiteren entsprechen, in solchen Beispielen, die Endpunkte, die für das Beaufsichtigungsnetzwerk 310 zugelassen werden sollen, individuellen Computereinrichtungen (z. B. Arbeitsstationen 308, den Servern 304 usw.), die ein IMM 230 implementieren. Zusätzlich oder alternativ entsprechen, in einigen Beispielen, die Endpunkte, die für das Beaufsichtigungsnetzwerk 310 zugelassen werden sollen, einem untergeordneten System für das Beaufsichtigungssystem auf der Steuerungssystemebene 106 (z. B. das Steuerungssystem 200). In einigen solcher Beispiele stellt das SIM 228, das in der primären Arbeitsstation implementiert ist, die zu dem untergeordneten System gehört (z. B. die primäre Arbeitsstation 220 des Steuerungssystems 200), Integritätsmessungen bereit, die durch das SIM 228 angefordert wurden, das in der Arbeitsstation des Beaufsichtigungssystems implementiert ist, wie das im Folgenden im Zusammenhang mit **Fig.** 16 beschrieben ist.

[0091] **Fig.** 14 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls 228 aus **Fig.** 2, 3 und/oder 6, um die Integrität eines Endpunkts zu verifizieren, der versucht, für ein Beaufsichtigungssystem zugelassen zu werden. In dem dargestellten Beispiel entspricht das SIM 228 einem SIM, das in einer primären Arbeitsstation des Beaufsichtigungssystems implementiert ist (z. B. der primären Arbeitsstation 306 des Beaufsichtigungssystems 300). Das beispielhafte Verfahren aus **Fig.** 14 beginnt in Block 1400, wo die beispielhafte Kommunikationsschnittstelle 602 eine Anforderung zur Verbindung von einem Endpunkt in dem Beaufsichtigungssystem empfängt. In einigen Beispielen kommt die Anforderung von einem bestimmten Endpunkt (z. B. einer Arbeitsstation 308, einem Server 304 usw.), der kommunikationsfähig mit dem Beaufsichtigungsnetzwerk des Beaufsichtigungssystems gekoppelt ist. In einigen Beispielen kommt die Anforderung von einer primären Arbeitsstation eines untergeordneten Systems (z. B. der primären Arbeitsstation 220 des Steuerungssystems 200). In Block 1402 sendet die beispielhafte Kommunikationsschnittstelle 602 eine Nonce (z. B. erzeugt durch die beispielhafte Autorisierungssteuerungseinrichtung 606) an den Endpunkt zusammen mit einer Anforderung für Integritätsmessungen für den Endpunkt.

[0092] In Block 1404 empfängt die beispielhafte Kommunikationsschnittstelle 602 einen Bericht der Integritätsmessungen. In einigen Beispielen werden die Integritätsmessungen für einen Endpunkt auf der Beaufsichtigungssystemebene 108 durch das IMM erzeugt, das durch den Endpunkt in gleicher oder ähnlicher Weise implementiert ist, wie das oben für ein IMM beschrieben wurde, das durch einen Endpunkt auf der Steuerungssystemebene 106 implementiert ist. Allerdings verfügt ein Beaufsichtigungssystem üblicherweise nicht über eine speziell ausgewiesene Konfiguration, wie das bei einem Steuerungssystem der Fall ist. Dementsprechend umfassen, in einigen Beispielen, die berichteten Integritätsmessungen keine Konfigurationsintegritätsmessungen. Andererseits können die Integritätsmessungen, in einigen Beispielen, wo der Endpunkt einem Steuerungssystem entspricht, einen Wert umfassen, welcher der Kombination aus Konfigurationsintegritätsmessungen von allen Endpunkten innerhalb des Steuerungssystems entspricht, wie das im Folgenden im Zusammenhang mit **Fig.** 16 ausführlicher beschrieben wird. In Block 1406 speichert die beispielhafte Integritätsmessungsdatenbank 618 die berichteten Integritätsmessungen.

[0093] In Block 1408 vergleicht der beispielhafte Integritätsmessungskomparator 604 die berichteten Integritätsmessungen mit Referenzwerten. In Block 1410 ermittelt der beispielhafte Integritätsmessungskomparator 604, ob die berichteten Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn das nicht der Fall ist, wird die Steuerung mit Block 1412 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 einen Administrator auffordert, die Diskrepanz zu überschreiben. In Block 1414 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob die Überschreibung bestätigt ist. Wenn das der Fall ist, wird die Steuerung mit Block 1416 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die berichteten Integritätsmessungen als die neuen Referenzwerte zuweist. In Block 1418 autorisiert die beispielhafte Autorisierungssteuerungseinrichtung 606 den Endpunkt zum vollständigen Kommunikationszugriff, indem sie dem Endpunkt Autorisierungsinformationen bereitstellt. Wenn wiederum in Block 1410 der beispielhafte Integritätsmessungskomparator 604 ermittelt, dass die berichteten Integritätsmessungen mit den Referenzwerten übereinstimmen, wird die Steuerung direkt mit Block 1418 fortgesetzt. In Block 1420 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob es einen weiteren Endpunkt gibt. Wenn das der Fall ist, kehrt die Steuerung zu Block 1400 zurück. Andernfalls endet das beispielhafte Verfahren aus **Fig.** 14.

[0094] Wenn wiederum in Block 1414 die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass die Überschreibung nicht bestätigt ist (z. B. wenn der Administrator die berichtete Integritätsmessung nicht als den neuen Referenzwert verwenden möchte), wird die Steuerung mit Block 1422 fortgesetzt. In Block 1422 schränkt die beispielhafte Autorisierungssteuerungseinrichtung 606 den Kommunikationszugriff des Endpunkts ein. In einigen Beispielen wird der Kommunikationszugriff abgelehnt (d. h. es wird kein Kommunikationszugriff gewährt). In einigen Beispielen wird der Endpunkt auf den Kommunikationszugriff für Firmware-Updates einschränkt. In anderen Beispielen kann die Kommunikation ermöglicht werden, aber die Daten von dem konkreten Knoten werden als verdächtig markiert oder gekennzeichnet. In einigen Beispielen kann der Kommunikationszugriff zu bestimmten Endpunkten in dem Beaufsichtigungssystem gewährt werden, während die Kommunikation zu anderen Endpunkten verhindert wird. Nach dem Einschränken des Kommunikationszugriffs für den Endpunkt (Block 1422) wird die Steuerung mit Block 1420 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung ermittelt, ob es einen weiteren Endpunkt wie oben beschrieben gibt.

[0095] **Fig.** 15 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls 228 aus **Fig.** 2, 3 und/oder 6, um die Integrität eines Endpunkts in einem Beaufsichtigungsnetzwerk eines Beaufsichtigungssystems während der Laufzeit zu überprüfen. Das heißt, während **Fig.** 14 ein beispielhaftes Verfahren repräsentiert, um die Integrität eines Endpunkts zu verifizieren, der erstmalig für das Beaufsichtigungssystems zugelassen wird, repräsentiert **Fig.** 15 die Verifizierung der Integrität des Endpunkts zu einem späteren Zeitpunkt, während sich der Endpunkt im Betrieb und/oder im Dienst befindet. Auf diese Weise können potenzielle Sicherheitsverstöße erkannt werden, die auftreten, nachdem der Endpunkt erworben, installiert und zur Verwendung konfiguriert wurde, und es kann entsprechend darauf reagiert werden. Wie oben beschrieben wurde, entspricht das in **Fig.** 15 dargestellte Beispiel des SIM 228 einem SIM, das in einer primären Arbeitsstation eines Beaufsichtigungssystems implementiert ist (z. B. der primären Arbeitsstation 306 des Beaufsichtigungssystems 300).

[0096] Das beispielhafte Verfahren aus **Fig.** 15 beginnt in Block 1502, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, ob es an der Zeit ist, die Laufzeitintegrität eines Endpunkts zu verifizieren. In einigen Beispielen werden die Laufzeit-Integritätsmessungen in einem geeigneten periodischen oder aperiodischen

Zeitplan durchgeführt. Zum Beispiel können die Laufzeit-Integritätsmessungen für jeden Endpunkt in einem Netzwerk einmal täglich angefordert werden. Darüber hinaus können, in einigen Beispielen, bestimmte Endpunkte häufiger als andere getestet werden, was sich zum Beispiel danach richtet, wie kritisch der Endpunkt für das Beaufsichtigungssystem ist, in dem der Endpunkt betrieben wird. Wenn die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass es nicht an der Zeit ist, die Laufzeitintegrität eines Endpunkts zu verifizieren (Block 1502), wartet die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, bis es an der Zeit ist.

[0097] Sobald es an der Zeit ist, die Laufzeitintegrität des Endpunkts zu verifizieren, wird die Steuerung mit Block 1504 fortgesetzt, wo die beispielhafte Kommunikationsschnittstelle 602 eine Nonce (z. B. erzeugt durch die beispielhafte Autorisierungssteuerungseinrichtung 606) an den Endpunkt zusammen mit einer Anforderung für Integritätsmessungen für den Endpunkt sendet. In Block 1506 empfängt die beispielhafte Kommunikationsschnittstelle 602 einen Bericht der Integritätsmessungen. In einigen Beispielen werden die Integritätsmessungen für einen Endpunkt auf der Beaufsichtigungssystemebene 108 durch das IMM erzeugt, das durch den Endpunkt in gleicher oder ähnlicher Weise implementiert ist, wie das oben für ein IMM beschrieben wurde, das durch einen Endpunkt auf der Steuerungssystemebene 106 implementiert ist. Allerdings umfassen, in einigen Beispielen, die Integritätsmessungen keine Konfigurationsintegritätsmessung, weil Beaufsichtigungssysteme üblicherweise nicht über Konfigurationsdateien verfügen, die definieren, wie das System konfiguriert werden soll. Andererseits können die Integritätsmessungen, in einigen Beispielen, wo der Endpunkt einem Steuerungssystem entspricht, Werte umfassen, welcher der Kombination aus Konfigurationsintegritätsmessungen von allen Endpunkten innerhalb des Steuerungssystems entspricht, wie das im Folgenden im Zusammenhang mit **Fig.** 16 ausführlicher beschrieben wird. In Block 1508 speichert die beispielhafte Integritätsmessungsdatenbank 618 die berichteten Integritätsmessungen.

[0098] In Block 1510 vergleicht der beispielhafte Integritätsmessungskomparator 604 die berichteten Integritätsmessungen mit Referenzwerten. In Block 1512 ermittelt der beispielhafte Integritätsmessungskomparator 604, ob die berichteten Integritätsmessungen mit den Referenzwerten übereinstimmen. Wenn das nicht der Fall ist, wird die Steuerung mit Block 1514 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 basierend auf den fehlgeschlagenen Integritätsmessungen eine Reaktion implementiert. In einigen Beispielen hängt die konkrete Reaktion davon ab, welcher Typ von Integritätsmessungen (z. B. Software-, Hardware- und/oder Konfiguration) nicht mit den Referenzwerten übereinstimmt. In einigen Beispielen hängt die Art und Weise der Reaktion von der Art und Weise des Endpunkts und/oder seiner Beziehung zu anderen Endpunkten in dem entsprechenden Netzwerk ab. In einigen Beispielen umfasst die Reaktion das Erzeugen einer Warnung (Block 1516), das Markieren künftiger Daten von dem Endpunkt als verdächtig (Block 1518) und/oder das Einschränken des Kommunikationszugriffs für den Endpunkt (Block 1520). In einigen Beispielen wird der Kommunikationszugriff abgelehnt (d. h. es wird kein Kommunikationszugriff gewährt). In einigen Beispielen wird der Endpunkt auf den Kommunikationszugriff für Firmware-Updates einschränkt. In anderen Beispielen kann die Kommunikation ermöglicht werden, aber die Daten von dem konkreten Knoten werden als verdächtig markiert oder gekennzeichnet. In einigen Beispielen kann der Kommunikationszugriff zu bestimmten Endpunkten in dem Beaufsichtigungssystem gewährt werden, während die Kommunikation zu anderen Endpunkten verhindert wird. In einigen Beispielen umfasst die Reaktion eine Kombination aus mehr als einem der Blöcke 156, 158, 1520 und 1522. In einigen Beispielen wird die implementierte Reaktion durch einen Benutzer definiert, wenn der Endpunkt konfiguriert wird.

[0099] Nachdem die entsprechende Reaktion implementiert wurde (Block 1514), wird die Steuerung mit Block 1522 fortgesetzt. Wenn wiederum in Block 1512 der beispielhafte Integritätsmessungskomparator 604 ermittelt, dass die Integritätsmessungen mit den Referenzwerten übereinstimmen, ist die Integrität des Endpunkts verifiziert, und keine weiteren Aktionen sind erforderlich. In solchen Beispielen wird die Steuerung direkt mit Block 1522 fortgesetzt. In Block 1522 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob es einen weiteren Endpunkt gibt, der verifiziert werden muss. Wenn das der Fall ist, kehrt die Steuerung zu Block 1502 zurück. Andernfalls wird die Steuerung mit Block 1526 fortgesetzt, wo die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, ob die Steuerung fortgesetzt werden soll. Wenn das der Fall ist, kehrt die Steuerung zu Block 1502 zurück. Andernfalls endet das beispielhafte Verfahren aus **Fig.** 15.

[0100] **Fig.** 16 ist ein Ablaufdiagramm zur Darstellung eines beispielhaften Verfahrens zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls (System Integrity Monitor - SIM) 228 aus **Fig.** 2, 3 und/oder 6 zur Erzeugung kombinierter Integritätsmessungen für einen Endpunkt, der einem System von auf niedrigeren Ebenen angeordneten Endpunkten (z. B. dem Steuerungssystem 200 als ein Endpunkt in dem Beaufsichtigungssystem 300 aus **Fig.** 3) entspricht. Das heißt, in dem dargestell-

ten Beispiel aus **Fig.** 16 entspricht das SIM 228 einem SIM, das in einer primären Arbeitsstation eines Steuerungssystems implementiert ist (z. B. der primären Arbeitsstation 220 des Steuerungssystems 200) und das mit einer primären Arbeitsstation eines Beaufsichtigungsnetzwerks kommuniziert (z. B. mit der primären Arbeitsstation 306 des Beaufsichtigungssystems 300). Anders ausgedrückt: Aus Sicht der primären Arbeitsstation 306 des Beaufsichtigungssystems 300 hat das in **Fig.** 6 implementierte SIM 228 die Aufgabe, Integritätsmessungen in gleicher oder ähnlicher Weise zu berichten wie die oben beschriebenen IMMs 230. Allerdings muss, wie das im Folgenden ausführlicher beschrieben wird, das SIM 228 zum Erzeugen solcher Integritätsmessungen zunächst Integritätsmessungen von den IMMs 230 erhalten oder erfassen, die in auf niedrigeren Ebenen angeordneten Endpunkten implementiert sind und vom SIM 228 überwacht werden.

[0101] Das beispielhafte Verfahren aus **Fig.** 16 beginnt in Block 1602, wo die beispielhafte Kommunikationsschnittstelle 602 eine Nonce mit einer Anforderung von einem Systemintegritätsüberwachungsmodul eines Beaufsichtigungssystems für Integritätsmessungen für einen Endpunkt empfängt, der einem untergeordneten System entspricht (z. B. einem Steuerungssystem innerhalb des Beaufsichtigungssystems). In einigen Beispielen ist, obwohl der Endpunkt sich auf ein gesamtes Subsystem bezieht, der Endpunkt effektiv durch eine primäre Arbeitsstation in dem Subsystem repräsentiert, welches das SIM 228 implementiert, das die Anforderung empfängt. In einigen Beispielen ist der zusammen mit der Anforderung bereitgestellte Nonce-Wert bei der Erzeugung der Antwort zu verwenden. In Block 1604 testet die beispielhafte Integritätsmessungs-Selbsttesteinrichtung 504 die Integritätsmessungsfunktionen des SIM. In Block 1606 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob die Integritätsmessungsfunktionen ordnungsgemäß funktionieren (z. B. ob die Ausgaben erwartungsgemäß sind). Wenn das nicht der Fall ist, endet das beispielhafte Verfahren aus **Fig.** 16. In einigen Beispielen erzeugt die Integritätsüberwachungsmodul-Steuerungseinrichtung 612 eine Fehlermeldung, die darüber informiert, dass der Test fehlgeschlagen ist, bevor das Verfahren endet. Wenn die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass die Integritätsmessungsfunktionen ordnungsgemäß funktionieren (Block 1606), wird die Steuerung mit Block 1608 fortgesetzt.

[0102] In Block 1608 löscht die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 alle der Integritätsmessungsregister 622, 624, 626. In Block 1610 fügt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die Software-Integritätsmessung von einem Endpunkt in dem untergeordneten System zum beispielhaften Software-Messungsregister 622 hinzu. In einigen Beispielen wird die Software-Integritätsmessung von der Integritätsmessungsdatenbank 618 gewonnen, die zuvor eine Anforderung von dem SIM 228 empfangen hat, das den entsprechenden Endpunkt überwacht. In anderen Beispielen fordert das SIM 228 die Software-Integritätsmessung als Antwort auf die in Block 1602 empfangene Anforderung an. In Block 1612 fügt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die Hardware-Integritätsmessung von dem Endpunkt in dem untergeordneten System zum beispielhaften Hardware-Messungsregister 624 hinzu. In einigen Beispielen wird die Hardware-Integritätsmessung von der Integritätsmessungsdatenbank 618 gewonnen, die zuvor eine Anforderung von dem SIM 228 empfangen hat, das den entsprechenden Endpunkt überwacht. In anderen Beispielen fordert das SIM 228 die Hardware-Integritätsmessung als Antwort auf die in Block 1602 empfangene Anforderung an. In Block 1614 fügt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 die Konfigurationsintegritätsmessung von dem Endpunkt in dem untergeordneten System zum beispielhaften Konfigurationsmessungsregister 626 hinzu. In einigen Beispielen wird die Konfigurationsintegritätsmessung von der Integritätsmessungsdatenbank 618 gewonnen, die zuvor eine Anforderung von dem SIM 228 empfangen hat, das den entsprechenden Endpunkt überwacht. In anderen Beispielen fordert das SIM 228 die Konfigurationsintegritätsmessung als Antwort auf die in Block 1602 empfangene Anforderung an.

[0103] In Block 1616 ermittelt die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612, ob es einen weiteren Endpunkt in dem untergeordneten System gibt. Wenn das der Fall ist, kehrt die Steuerung zu den Blöcken 1610, 1612 und 1614 zurück, um die entsprechenden Integritätsmessungen von dem Endpunkt zu den entsprechenden Registern 622, 624, 626 hinzuzufügen. In einigen Beispielen werden die Integritätsmessungen zu den jeweiligen Registern hinzugefügt, indem sie an zuvor hinzugefügte Integritätsmessungen angehängt werden. Wenn die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612 ermittelt, dass es keine weiteren Endpunkte in dem untergeordneten System gibt (Block 1616), wird die Steuerung mit Block 1618 fortgesetzt, wo der beispielhafte Integritätsmessungsgenerator 608 kombinierte Integritätsmessungen für das untergeordnete System erzeugt. In einigen Beispielen werden die kombinierten Integritätsmessungen erzeugt, indem eine Prüfsumme der Zeichenfolge von Werten in jedem der Software-, Hardware- und Konfigurationsregister 622, 624, 626 berechnet wird. In Block 1620 sendet die beispielhafte Kommunikationsschnittstelle 602 einen Bericht der kombinierten Integritätsmessungen

an das SIM des Beaufsichtigungssystems, woraufhin das beispielhafte Verfahren aus **Fig.** 16 endet.

[0104] **Fig.** 17 ist ein Blockdiagramm einer beispielhaften Prozessorplattform 1700, die verwendet und/oder programmiert werden kann, um das beispielhafte Verfahren aus **Fig.** 7 zur Implementierung des beispielhaften Konfigurationsmoduls 226 aus **Fig.** 2 und/oder 4 auszuführen. Die Prozessorplattform 1700 kann zum Beispiel ein Server, ein Personal Computer, ein mobiles Gerät (z. B. ein Mobiltelefon, ein Smartphone, ein Tablet wie etwa ein iPad™), ein Personal Digital Assistant (PDA), ein Internet-Gerät, ein DVD-Abspielgerät, ein CD-Abspielgerät, ein digitaler Videorekorder, ein Blu-ray-Abspielgerät, eine Spielkonsole, ein persönlicher Videorekorder, eine Set-Top-Box oder jeder andere Typ von Computereinrichtung sein.

[0105] Die Prozessorplattform 1700 des dargestellten Beispiels umfasst einen Prozessor 1712. Der Prozessor 1712 des dargestellten Beispiels ist Hardware. Beispielsweise kann der Prozessor 1712 durch einen oder mehrere integrierte Schaltungen, Logikschaltungen, Mikroprozessoren oder Steuerungseinrichtungen aus jeder gewünschten Familie und von jedem Hersteller implementiert sein.

[0106] Der Prozessor 1712 des dargestellten Beispiels umfasst einen lokalen Speicher 1713 (z. B. einen Zwischenspeicher). In dem dargestellten Beispiel implementiert der Prozessor 1712 den beispielhaften Konfigurationsdateigenerator 404 und/oder den beispielhaften Referenzwertgenerator 406. Der Prozessor 1712 des dargestellten Beispiels kommuniziert über einen Bus 1718 mit einem Hauptspeicher, welcher einen flüchtigen Speicher 1714 und einen nicht-flüchtigen Speicher 1716 umfasst. Die Implementierung des flüchtigen Speichers 1714 kann durch synchronen dynamischen Direktzugriffsspeicher (Synchronous Dynamic Random Access Memory - SDRAM), dynamischen Direktzugriffsspeicher (Dynamic Random Access Memory - DRAM), dynamischen RAMBUS-Direktzugriffsspeicher (RAMBUS Dynamic Random Access Memory - RDRAM) und/oder jeden anderen Typ von Direktzugriffsspeichereinrichtung erfolgen. Die Implementierung des nicht-flüchtigen Speichers 1716 kann durch Flash-Speicher und/oder jeden anderen Typ von Speichereinrichtung erfolgen. Der Zugriff auf den Hauptspeicher 1714, 1716 wird durch eine Speichersteuerungseinrichtung gesteuert.

[0107] Die Prozessorplattform 1700 des dargestellten Beispiels umfasst auch eine Schnittstellenschaltung 1720. Die Implementierung der Schnittstellenschaltung 1720 kann durch jeden Typ von Schnittstellenstandard erfolgen, z. B. durch eine Ethernet-Schnittstelle, einen Universal Serial Bus (USB) und/oder eine PCI-Express-Schnittstelle.

[0108] In dem dargestellten Beispiel sind eine oder mehrere Eingabeeinrichtungen 1722 mit der Schnittstellenschaltung 1720 verbunden. Die Eingabeeinrichtungen 1722 ermöglichen einem Benutzer die Eingabe von Daten und Befehlen in den Prozessor 1712. Die Implementierung der Eingabeeinrichtungen kann zum Beispiel durch einen Audiosensor, ein Mikrofon, eine Kamera (Standbild oder Video), eine Tastatur, eine Taste, eine Maus, einen Touchscreen, ein Trackpad, einen Trackball, Isopoint und/oder ein Spracherkennungssystem erfolgen.

[0109] Außerdem sind mit der Schnittstellenschaltung 1720 des dargestellten Beispiels eine oder mehrere Ausgabeeinrichtungen 1724 verbunden. Die Implementierung der Ausgabeeinrichtungen 1724 kann zum Beispiel durch Anzeigeeinrichtungen (z. B. eine Leuchtdiode (Light Emitting Diode - LED), eine organische Leuchtdiode (Organic Light Emitting Diode - OLED), eine Flüssigkristallanzeige, eine Anzeige mit Kathodenstrahlröhre (Cathode Ray Tube - CRT), einen Touchscreen, eine taktile Ausgabeeinrichtung, eine Leuchtdiode (Light Emitting Diode - LED), einen Drucker und/oder Lautsprecher) erfolgen. Die Schnittstellenschaltung 1720 des dargestellten Beispiels umfasst somit üblicherweise eine Grafiktreiberkarte, einen Grafiktreiberchip oder einen Grafiktreiberprozessor.

[0110] Die Schnittstellenschaltung 1720 des dargestellten Beispiels umfasst auch eine Kommunikationseinrichtung wie z. B. einen Sender, einen Empfänger, einen Transceiver, ein Modem und/oder eine Netzwerkschnittstellenkarte, um den Austausch von Daten mit externen Maschinen (z. B. Computereinrichtungen jeglicher Art) über ein Netzwerk 1726 (z. B. eine Ethernet-Verbindung, eine digitale Teilnehmerleitung (Digital Subscriber Line - DSL), eine Telefonleitung, ein Koaxialkabel, ein Mobiltelefonsystem usw.) zu ermöglichen.

[0111] Die Prozessorplattform 1700 des dargestellten Beispiels umfasst auch eine oder mehrere Massenspeichereinrichtungen 1728 zum Speichern von Software und/oder Daten. Zum Beispiel kann die Massenspeichereinrichtung 1728 die beispielhafte Konfigurationsdatenbank 408 aus **Fig.** 4 umfassen. Beispiele solcher Massenspeichereinrichtungen 1728 sind Diskettenlaufwerke, Festplatten, Compact-Disk-Laufwerke. Blu-ray-Disk-Laufwerke, RAID-Systeme und DVD-Laufwerke (Digital Versatile Disk).

[0112] Codierte Anweisungen 1732 zur Implementierung des Verfahrens aus **Fig.** 7 können in der Massenspeichereinrichtung 1728, in dem flüchtigen Speicher 1714, in dem nicht-flüchtigen Speicher 1716 und/oder auf einem entnehmbaren greifbaren computerlesbaren Speichermedium wie z. B. einer CD oder DVD gespeichert sein.

[0113] **Fig.** 18 ist ein Blockdiagramm einer beispielhaften Prozessorplattform 1800, die verwendet und/oder programmiert werden kann, um die beispielhaften Verfahren aus **Fig.** 8-10 zur Implementierung des beispielhaften Integritätsmessungsmoduls 230 aus **Fig.** 2, 3 und/oder 5 auszuführen. Die Prozessorplattform 1800 kann zum Beispiel ein Server, ein Personal Computer, ein mobiles Gerät (z. B. ein Mobiltelefon, ein Smartphone, ein Tablet wie etwa ein iPad™), ein Personal Digital Assistant (PDA), ein Internet-Gerät, ein DVD-Abspielgerät, ein CD-Abspielgerät, ein digitaler Videorekorder, ein Blu-ray-Abspielgerät, eine Spielkonsole, ein persönlicher Videorekorder, eine Set-Top-Bock oder jeder andere Typ von Computereinrichtung sein.

[0114] Die Prozessorplattform 1800 des dargestellten Beispiels umfasst einen Prozessor 1812. Der Prozessor 1812 des dargestellten Beispiels ist Hardware. Beispielsweise kann der Prozessor 1812 durch einen oder mehrere integrierte Schaltungen, Logikschaltungen, Mikroprozessoren oder Steuerungseinrichtungen aus jeder gewünschten Familie und von jedem Hersteller implementiert sein.

[0115] Der Prozessor 1812 des dargestellten Beispiels umfasst einen lokalen Speicher 1813 (z. B. einen Zwischenspeicher). In dem dargestellten Beispiel implementiert der Prozessor 1812 den beispielhaften Integritätsmessungsgenerator 502, die beispielhafte Integritätsmessungs-Selbsttesteinrichtung 504, die beispielhafte Integritätsmessungs-Steuerungseinrichtung 506 und/oder den beispielhaften Kommunikationsauthentifikator 508. Der Prozessor 1812 des dargestellten Beispiels kommuniziert über einen Bus 1818 mit einem Hauptspeicher, welcher einen flüchtigen Speicher 1814 und einen nicht-flüchtigen Speicher 1816 umfasst. Die Implementierung des flüchtigen Speichers 1814 kann durch synchronen dynamischen Direktzugriffsspeicher (Synchronous Dynamic Random Access Memory - SDRAM), dynamischen Direktzugriffsspeicher (Dynamic Random Access Memory - DRAM), dynamischen RAMBUS-Direktzugriffsspeicher (RAMBUS Dynamic Random Access Memory - RDRAM) und/oder jeden anderen Typ von Direktzugriffsspeichereinrichtung erfolgen. Die Implementierung des nicht-flüchtigen Speichers 1816 kann durch Flash-Speicher und/oder jeden anderen Typ von Speichereinrichtung erfolgen. Der Zugriff auf den Hauptspeicher 1814, 1816 wird durch eine Speichersteuerungseinrichtung gesteuert.

[0116] Die Prozessorplattform 1800 des dargestellten Beispiels umfasst auch eine Schnittstellenschaltung 1820. Die Implementierung der Schnittstellenschaltung 1820 kann durch jeden Typ von Schnittstellenstandard erfolgen, z. B. durch eine Ethernet-Schnittstelle, einen Universal Serial Bus (USB) und/oder eine PCI-Express-Schnittstelle.

[0117] In dem dargestellten Beispiel sind eine oder mehrere Eingabeeinrichtungen 1822 mit der Schnittstellenschaltung 1820 verbunden. Die Eingabeeinrichtungen 1822 ermöglichen einem Benutzer die Eingabe von Daten und Befehlen in den Prozessor 1812. Die Implementierung der Eingabeeinrichtungen kann zum Beispiel durch einen Audiosensor, ein Mikrofon, eine Kamera (Standbild oder Video), eine Tastatur, eine Taste, eine Maus, einen Touchscreen, ein Trackpad, einen Trackball, Isopoint und/oder ein Spracherkennungssystem erfolgen.

[0118] Außerdem sind mit der Schnittstellenschaltung 1820 des dargestellten Beispiels eine oder mehrere Ausgabeeinrichtungen 1824 verbunden. Die Implementierung der Ausgabeeinrichtungen 1824 kann zum Beispiel durch Anzeigeeinrichtungen (z. B. eine Leuchtdiode (Light Emitting Diode - LED), eine organische Leuchtdiode (Organic Light Emitting Diode - OLED), eine Flüssigkristallanzeige, eine Anzeige mit Kathodenstrahlröhre (Cathode Ray Tube - CRT), einen Touchscreen, eine taktile Ausgabeeinrichtung, eine Leuchtdiode (Light Emitting Diode - LED), einen Drucker und/oder Lautsprecher) erfolgen. Die Schnittstellenschaltung 1820 des dargestellten Beispiels umfasst somit üblicherweise eine Grafiktreiberkarte, einen Grafiktreiberchip oder einen Grafiktreiberprozessor.

[0119] Die Schnittstellenschaltung 1820 des dargestellten Beispiels umfasst auch eine Kommunikationseinrichtung wie z. B. einen Sender, einen Empfänger, einen Transceiver, ein Modem und/oder eine Netzwerkschnittstellenkarte, um den Austausch von Daten mit externen Maschinen (z. B. Computereinrichtungen jeglicher Art) über ein Netzwerk 1826 (z. B. eine Ethernet-Verbindung, eine digitale Teilnehmerleitung (Digital Subscriber Line - DSL), eine Telefonleitung, ein Koaxialkabel, ein Mobiltelefonsystem usw.) zu ermöglichen.

[0120] Die Prozessorplattform 1800 des dargestellten Beispiels umfasst auch eine oder mehrere Massenspeichereinrichtungen 1828 zum Speichern von Software und/oder Daten. Zum Beispiel kann die Massenspeichereinrichtung 1928 [Anmerkung des Übersetzers: die Bezugsziffer müsste hier 1828 sein.] das beispielhafte Software-Messungsregister 512, das beispielhafte Konfigurationsmessungsregister 514, das beispielhafte Hardware-Messungsregister 516, die beispielhafte Autorisierungsinformationsdatenbank 518 und/oder die beispielhafte Konfigurationsdatenbank 520 umfassen. Beispiele solcher Massenspeichereinrichtungen 1828 sind Diskettenlaufwerke, Festplatten, Compact-Disk-Laufwerke. Blu-ray-Disk-Laufwerke, RAID-Systeme und DVD-Laufwerke (Digital Versatile Disk).

[0121] Codierte Anweisungen 1832 zur Implementierung des Verfahrens aus **Fig.** 8-10 können in der

Massenspeichereinrichtung 1828, in dem flüchtigen Speicher 1814, in dem nicht-flüchtigen Speicher 1816 und/oder auf einem entnehmbaren greifbaren computerlesbaren Speichermedium wie z. B. einer CD oder DVD gespeichert sein.

[0122] **Fig.** 19 ist ein Blockdiagramm einer beispielhaften Prozessorplattform 1900, die verwendet und/oder programmiert werden kann, um die beispielhaften Verfahren aus **Fig.** 11-16 zur Implementierung des beispielhaften Systemintegritätsüberwachungsmoduls 228 aus **Fig.** 2, 3 und/oder 6 auszuführen. Die Prozessorplattform 1900 kann zum Beispiel ein Server, ein Personal Computer, ein mobiles Gerät (z. B. ein Mobiltelefon, ein Smartphone, ein Tablet wie etwa ein iPad™), ein Personal Digital Assistant (PDA), ein Internet-Gerät, ein DVD-Abspielgerät, ein CD-Abspielgerät, ein digitaler Videorekorder, ein Blu-ray-Abspielgerät, eine Spielkonsole, ein persönlicher Videorekorder, eine Set-Top-Bock oder jeder andere Typ von Computereinrichtung sein.

[0123] Die Prozessorplattform 1900 des dargestellten Beispiels umfasst einen Prozessor 1912. Der Prozessor 1912 des dargestellten Beispiels ist Hardware. Beispielsweise kann der Prozessor 1912 durch einen oder mehrere integrierte Schaltungen, Logikschaltungen, Mikroprozessoren oder Steuerungseinrichtungen aus jeder gewünschten Familie und von jedem Hersteller implementiert sein.

[0124] Der Prozessor 1912 des dargestellten Beispiels umfasst einen lokalen Speicher 1913 (z. B. einen Zwischenspeicher). In dem dargestellten Beispiel implementiert der Prozessor 1912 den beispielhaften Integritätsmessungskomparator 604, die beispielhafte Autorisierungssteuerungseinrichtung 606, den beispielhaften Integritätsmessungsgenerator 608, die beispielhafte Integritätsmessungs-Selbsttesteinrichtung 610 und/oder die beispielhafte Integritätsüberwachungsmodul-Steuerungseinrichtung 612. Der Prozessor 1912 des dargestellten Beispiels kommuniziert über einen Bus 1918 mit einem Hauptspeicher, welcher einen flüchtigen Speicher 1914 und einen nicht-flüchtigen Speicher 1916 umfasst. Die Implementierung des flüchtigen Speichers 1914 kann durch synchronen dynamischen Direktzugriffsspeicher (Synchronous Dynamic Random Access Memory - SDRAM), dynamischen Direktzugriffsspeicher (Dynamic Random Access Memory - DRAM), dynamischen RAMBUS-Direktzugriffsspeicher (RAMBUS Dynamic Random Access Memory - RDRAM) und/oder jeden anderen Typ von Direktzugriffsspeichereinrichtung erfolgen. Die Implementierung des nicht-flüchtigen Speichers 1916 kann durch Flash-Speicher und/oder jeden anderen Typ von Speichereinrichtung erfolgen. Der Zugriff auf den Hauptspeicher 1914, 1916 wird durch eine Speichersteuerungseinrichtung gesteuert.

[0125] Die Prozessorplattform 1900 des dargestellten Beispiels umfasst auch eine Schnittstellenschaltung 1920. Die Implementierung der Schnittstellenschaltung 1920 kann durch jeden Typ von Schnittstellenstandard erfolgen, z. B. durch eine Ethernet-Schnittstelle, einen Universal Serial Bus (USB) und/oder eine PCI-Express-Schnittstelle.

[0126] In dem dargestellten Beispiel sind eine oder mehrere Eingabeeinrichtungen 1922 mit der Schnittstellenschaltung 1920 verbunden. Die Eingabeeinrichtungen 1922 ermöglichen einem Benutzer die Eingabe von Daten und Befehlen in den Prozessor 1912. Die Implementierung der Eingabeeinrichtungen kann zum Beispiel durch einen Audiosensor, ein Mikrofon, eine Kamera (Standbild oder Video), eine Tastatur, eine Taste, eine Maus, einen Touchscreen, ein Trackpad, einen Trackball, Isopoint und/oder ein Spracherkennungssystem erfolgen.

[0127] Außerdem sind mit der Schnittstellenschaltung 1920 des dargestellten Beispiels eine oder mehrere Ausgabeeinrichtungen 1924 verbunden. Die Implementierung der Ausgabeeinrichtungen 1924 kann zum Beispiel durch Anzeigeeinrichtungen (z. B. eine Leuchtdiode (Light Emitting Diode - LED), eine organische Leuchtdiode (Organic Light Emitting Diode - OLED), eine Flüssigkristallanzeige, eine Anzeige mit Kathodenstrahlröhre (Cathode Ray Tube - CRT), einen Touchscreen, eine taktile Ausgabeeinrichtung, eine Leuchtdiode (Light Emitting Diode - LED), einen Drucker und/oder Lautsprecher) erfolgen. Die Schnittstellenschaltung 1920 des dargestellten Beispiels umfasst somit üblicherweise eine Grafiktreiberkarte, einen Grafiktreiberchip oder einen Grafiktreiberprozessor.

[0128] Die Schnittstellenschaltung 1920 des dargestellten Beispiels umfasst auch eine Kommunikationseinrichtung wie z. B. einen Sender, einen Empfänger, einen Transceiver, ein Modem und/oder eine Netzwerkschnittstellenkarte, um den Austausch von Daten mit externen Maschinen (z. B. Computereinrichtungen jeglicher Art) über ein Netzwerk 1926 (z. B. eine Ethernet-Verbindung, eine digitale Teilnehmerleitung (Digital Subscriber Line - DSL), eine Telefonleitung, ein Koaxialkabel, ein Mobiltelefonsystem usw.) zu ermöglichen.

[0129] Die Prozessorplattform 1900 des dargestellten Beispiels umfasst auch eine oder mehrere Massenspeichereinrichtungen 1928 zum Speichern von Software und/oder Daten. Zum Beispiel kann die Massenspeichereinrichtung 1928 die beispielhafte Referenzwertdatenbank 616, die beispielhafte Integritätsmessungsdatenbank 618, die beispielhafte Autorisierungsinformationsdatenbank 620, das beispielhafte Software-Messungsregister 622, das beispielhafte Hardware-Messungsregister 624 und/oder das beispielhafte Konfigurationsmessungsregister

626 aus **Fig.** 6 umfassen. Beispiele solcher Massenspeichereinrichtungen 1928 sind Diskettenlaufwerke, Festplatten, Compact-Disk-Laufwerke. Blu-ray-Disk-Laufwerke, RAID-Systeme und DVD-Laufwerke (Digital Versatile Disk).

[0130] Codierte Anweisungen 1932 zur Implementierung des Verfahrens aus **Fig.** 11-16 können in der Massenspeichereinrichtung 1928, in dem flüchtigen Speicher 1914, in dem nicht-flüchtigen Speicher 1916 und/oder auf einem entnehmbaren greifbaren computerlesbaren Speichermedium wie z. B. einer CD oder DVD gespeichert sein.

[0131] Aus den vorstehenden Ausführungen lässt sich leicht nachvollziehen, dass die oben offenbarten Verfahren, Vorrichtungen und Herstellungsartikel die Sicherheit und Zuverlässigkeit von Komponenten in einem Industrieunternehmenssystem verbessern. Im Unterschied zu bisherigen Ansätzen, bei denen die Integrität in Netzwerkkommunikationseinrichtungen (Endpunkten) basierend auf Qualitätszusicherungen von Originalgeräteherstellern angenommen wird, lässt sich mit den hier offenbarten Beispielen die Integrität solcher Endpunkte verifizieren. Des Weiteren kann eine solche Verifizierung durchgeführt werden, wenn ein Endpunkt erstmalig in einem bestimmten Netzwerk registriert und/oder zugelassen wird und/oder während der Laufzeit, wenn sich der Endpunkt in Betrieb befindet. Auf diese Weise können potenzielle Sicherheitsverstöße und/oder andere Fehler, welche die Integrität der Endpunkte kompromittieren, unabhängig vom Zeitpunkt ihres Auftretens erkannt werden, um sicherzustellen, dass die Kommunikation von solchen Endpunkten keine anderen Endpunkte beeinträchtigt, und um Abhilfemaßnahmen im Hinblick auf den kompromittierten Endpunkt ergreifen zu können. Außerdem ermöglichen die hier offenbarten Lehren das Roll-Up und/oder die Kombination von Integritätsmessungen für auf niedrigeren Ebenen angeordnete Endpunkte zur Verifizierung eines entsprechenden Systems, das einem Beaufsichtigungssystem auf höherer Ebene untergeordnet ist.

[0132] Obwohl hier bestimmte beispielhafte Verfahren, Vorrichtungen und Herstellungsartikel offenbart wurden, ist der Umfang des vorliegenden Patents nicht auf diese beschränkt. Vielmehr deckt das vorliegende Patent alle Verfahren, Vorrichtungen und Herstellungsartikel ab, die angemessen vom Umfang der Ansprüche des vorliegenden Patents eingeschlossen sind.

Patentansprüche

1. Vorrichtung, umfassend:
einen Integritätsmessungskomparator zum Vergleichen einer Integritätsmessung mit einem Referenzwert, wobei die Integritätsmessung durch einen ersten Endpunkt in einem ersten Netzwerk eines Industrieunternehmenssystems basierend auf einem Status des ersten Endpunkts erzeugt wird, wobei der Referenzwert einem vertrauenswürdigen Status des Endpunkts entspricht, wobei der Zustand des Endpunkts mindestens einem von Software oder Firmware entspricht, die auf den Endpunkt geladen ist, Konfigurationsdaten, die dem Endpunkt zugewiesen sind, oder einer Hardware oder Peripherie, die mit dem Endpunkt gekoppelt ist, entspricht, und wobei der erste Endpunkt einer Gruppe von zweiten Endpunkten in einem Teilnetz des ersten Netzes entspricht, wobei die Integritätsmessung einer Kombination von verschiedenen Integritätsmessungen entspricht, die der Gruppe von zweiten Endpunkten in dem Teilnetz entsprechen; und
eine Autorisierungssteuerungseinrichtung zum Ermöglichen des Kommunikationszugriffs für den Endpunkt auf das Netzwerk basierend auf dem Vergleich der Integritätsmessung mit dem Referenzwert.

2. Vorrichtung gemäß Anspruch 1, des Weiteren umfassend eine Integritätsmessungs-Steuerungseinrichtung zum Aktualisieren des Referenzwerts zur Übereinstimmung mit der Integritätsmessung basierend auf einer bestätigten Überschreibung einer Diskrepanz zwischen der Integritätsmessung und dem Referenzwert; und/oder wobei der Kommunikationszugriff für den ersten Endpunkt dem vollständigen Kommunikationszugriff entspricht, wenn die Integritätsmessung mit dem Referenzwert übereinstimmt, wobei die Autorisierungssteuerungseinrichtung Autorisierungsinformationen an den ersten Endpunkt bereitstellt, um den vollständigen Kommunikationszugriff zu ermöglichen.

3. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 2, wobei die Integritätsmessung erzeugt wird, indem eine Prüfsumme der auf den ersten Endpunkt geladenen Software berechnet wird.

4. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 3, wobei der Kommunikationszugriff dem Kommunikationszugriff für Firmware-Updates entspricht, wenn die Prüfsumme nicht mit dem Referenzwert übereinstimmt.

5. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 2, wobei die Integritätsmessung erzeugt wird, indem eine Prüfsumme der auf den ersten Endpunkt gespeicherten Konfigurationsdaten berechnet wird.

6. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 5, wobei der Kommunikationszugriff dem Kommunikationszugriff zur Konfiguration entspricht, wenn die

Prüfsumme nicht mit dem Referenzwert übereinstimmt.

7. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 5, wobei der Referenzwert basierend auf einer für das erste Netzwerk erzeugten Konfigurationsdatei erzeugt wird.

8. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 2, wobei die Integritätsmessung erzeugt wird, indem eine Prüfsumme für die Peripherieeinrichtung des ersten Endpunkts berechnet wird.

9. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 1, wobei die Integritätsmessung durch den ersten Endpunkt während der Boot-Zeit des ersten Endpunkts erzeugt wird; und/oder wobei die Integritätsmessung durch den ersten Endpunkt während der Laufzeit des ersten Endpunkts erzeugt wird.

10. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 1, wobei die Autorisierungssteuerungseinrichtung mindestens eines des Folgenden durchführt:
(1) Erzeugen eines Alarms, wenn die Integritätsmessung nicht mit dem Referenzwert übereinstimmt, wobei der Alarm aussagekräftig in Bezug auf eine verdächtige Identität des ersten Endpunkts ist, oder
(2) Markieren der von dem ersten Endpunkt bereitgestellten Daten, wenn die Integritätsmessung nicht mit dem Referenzwert übereinstimmt, wobei die markierten Daten aussagekräftig in Bezug auf eine verdächtige Identität des ersten Endpunkts sind; und/oder
wobei die Autorisierungssteuerungseinrichtung den Kommunikationszugriff für den Endpunkt auf das Netzwerk verhindert und den Kommunikationszugriff für einen redundanten Endpunkt ermöglicht, wenn die Integritätsmessung nicht mit dem Referenzwert übereinstimmt.

11. Vorrichtung nach einem der vorhergehenden Ansprüche, insbesondere gemäß Anspruch 1, wobei das erste Netzwerk einem Steuerungsnetzwerk eines Steuerungssystems innerhalb des Industrieunternehmenssystems entspricht, wobei der erste Endpunkt einem des Folgenden entspricht: einer Arbeitsstation, einem Server, einer Steuerungseinrichtung, einer Eingabe-/Ausgabeeinrichtung oder einer Feldeinrichtung mit Kopplung an das Steuerungsnetzwerk; und/oder wobei das erste Netzwerk einem Beaufsichtigungsnetzwerk eines Beaufsichtigungssystems innerhalb des Industrieunternehmenssystems entspricht, wobei der erste Endpunkt einem Steuerungssystem entspricht, das dem Beaufsichtigungssystem untergeordnet ist.

12. Verfahren, umfassend:
Empfangen einer Integritätsmessung von einem ersten Endpunkt in einem ersten Netzwerk eines Industrieunternehmenssystems, wobei die Integritätsmessung durch den ersten Endpunkt basierend auf einem Status des ersten Endpunkts erzeugt wird;
Vergleichen der Integritätsmessung mit einem Referenzwert, der einem vertrauenswürdigen Status des Endpunkts entspricht, wobei der Zustand des Endpunkts mindestens einer Software oder Firmware, die auf dem Endpunkt geladen ist, Konfigurationsdaten, die dem Endpunkt zugewiesen sind, oder einer Hardware oder Peripherie, die mit dem Endpunkt gekoppelt ist, entspricht, und wobei der erste Endpunkt einer Gruppe von zweiten Endpunkten in einem Teilnetz des ersten Netzes entspricht, wobei die Integritätsmessung einer Kombination von verschiedenen Integritätsmessungen entspricht, die der Gruppe von zweiten Endpunkten in dem Teilnetz entsprechen; und
Ermöglichen des Kommunikationszugriffs für den ersten Endpunkt auf das erste Netzwerk basierend auf dem Vergleich.

13. Verfahren gemäß Anspruch 12, des Weiteren umfassend:
Empfangen einer Zulassung zum Überschreiben einer Diskrepanz zwischen der Integritätsmessung und dem Referenzwert; und
Aktualisieren des Referenzwerts zur Übereinstimmung mit der Integritätsmessung basierend auf der Zulassung; und/oder
des Weiteren umfassend das Ermöglichen des vollständigen Kommunikationszugriffs für den ersten Endpunkt, wenn die Integritätsmessung mit dem Referenzwert übereinstimmt, indem Autorisierungsinformationen an den ersten Endpunkt bereitgestellt werden.

14. Verfahren nach Anspruch 12 oder 13, insbesondere gemäß Anspruch 16, wobei die Integritätsmessung erzeugt wird, indem eine Prüfsumme der auf den ersten Endpunkt geladenen Software berechnet wird.

15. Verfahren nach einem der Ansprüche 12-14, insbesondere gemäß Anspruch 17, des Weiteren umfassend das Ermöglichen des Kommunikationszugriffs für Firmware-Updates, wenn die Prüfsumme nicht mit dem Referenzwert übereinstimmt.

16. Verfahren nach einem der Ansprüche 12-15, insbesondere gemäß Anspruch 16, wobei die Integritätsmessung erzeugt wird, indem der erste Endpunkt eine Prüfsumme der auf den ersten Endpunkt gespeicherten Konfigurationsdaten berechnet.

17. Verfahren nach einem der Ansprüche 12-16, insbesondere gemäß Anspruch 19, des Weiteren

umfassend das Ermöglichen des Kommunikationszugriffs zur Konfiguration, wenn die Prüfsumme nicht mit dem Referenzwert übereinstimmt; und/oder wobei der Referenzwert basierend auf einer für das erste Netzwerk erzeugten Konfigurationsdatei erzeugt wird.

18. Verfahren nach einem der Ansprüche 12-17, insbesondere gemäß Anspruch 16, wobei die Integritätsmessung durch den ersten Endpunkt erzeugt wird, indem eine Prüfsumme für die Peripherieeinrichtung des ersten Endpunkts berechnet wird.

19. Verfahren nach einem der Ansprüche 12-18, insbesondere gemäß Anspruch 14, wobei die Integritätsmessung durch den ersten Endpunkt während der Boot-Zeit des ersten Endpunkts erzeugt wird; und/oder
wobei die Integritätsmessung durch den ersten Endpunkt während der Laufzeit des ersten Endpunkts erzeugt wird; und/oder
des Weiteren mindestens umfassend eines des Folgenden:
Erzeugen eines Alarms, wenn die Integritätsmessung nicht mit dem Referenzwert übereinstimmt; oder
Markieren der von dem ersten Endpunkt bereitgestellten Daten, wenn die Integritätsmessung nicht mit dem Referenzwert übereinstimmt, wobei sowohl der Alarm als auch die markierten Daten aussagekräftig in Bezug auf eine verdächtige Identität des ersten Endpunkts sind.

20. Verfahren nach einem der Ansprüche 12-19, insbesondere gemäß Anspruch 12, des Weiteren umfassend:
Verhindern des Kommunikationszugriffs für den ersten Endpunkt auf das erste Netzwerk, wenn die Integritätsmessung nicht mit dem Referenzwert übereinstimmt; und
Ermöglichen des Kommunikationszugriffs für einen redundanten Endpunkt;
und/oder
wobei das erste Netzwerk einem Steuerungsnetzwerk eines Steuerungssystems innerhalb des Industrieunternehmenssystems entspricht, wobei der erste Endpunkt einem des Folgenden entspricht: einer Arbeitsstation, einem Server, einer Steuerungseinrichtung, einem Eingabe-/Ausgabeeinrichtung oder einer Feldeinrichtung mit Kopplung an das Steuerungsnetzwerk.

21. Verfahren nach einem der Ansprüche 12-20, insbesondere gemäß Anspruch 12, wobei das erste Netzwerk einem Beaufsichtigungsnetzwerk eines Beaufsichtigungssystems innerhalb des Industrieunternehmenssystems entspricht, wobei der erste Endpunkt einem Steuerungssystem entspricht, das dem Beaufsichtigungssystem untergeordnet ist.

22. Computerlesbares Speichermedium, das Anweisungen speichert, die bei Ausführung durch zumindest einen Prozessor bewirken, dass eine Maschine eines der Verfahren nach einem der Ansprüche 12-21 implementiert.

Es folgen 19 Seiten Zeichnungen

Anhängende Zeichnungen

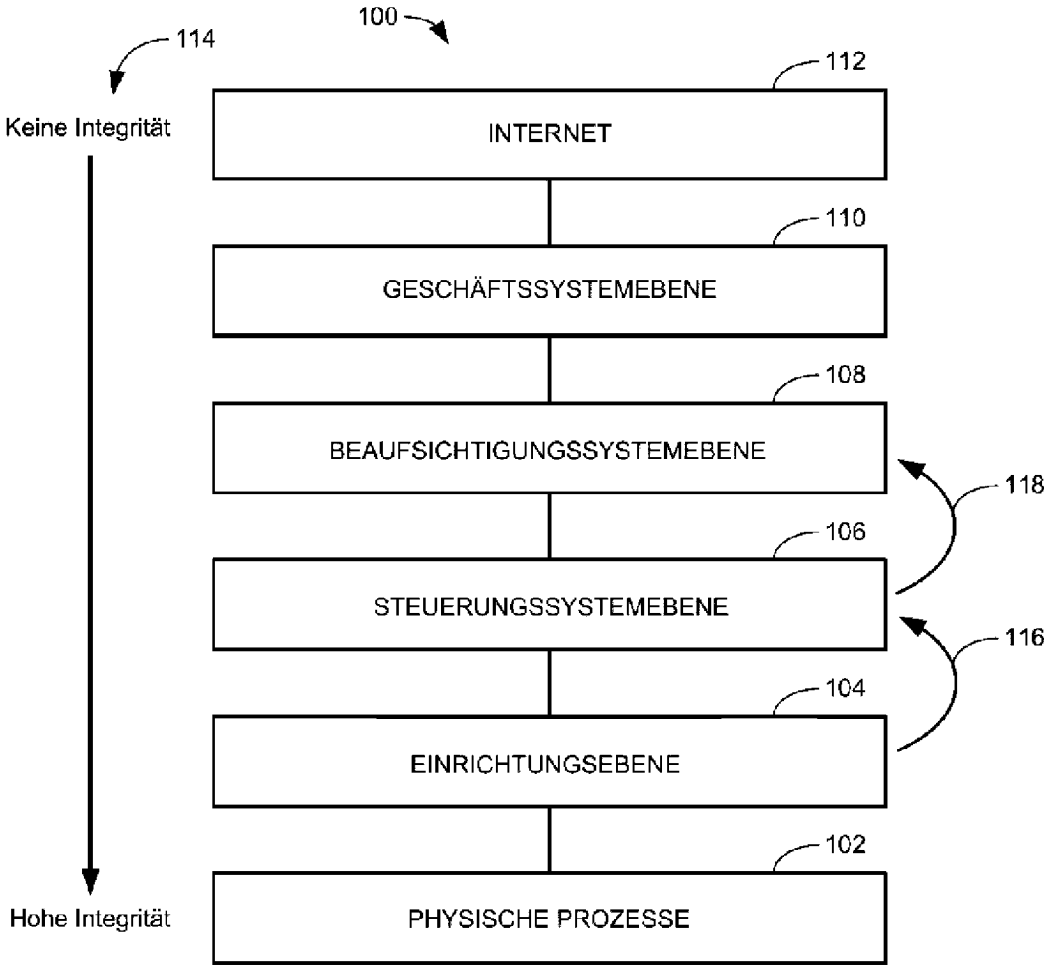


FIG. 1

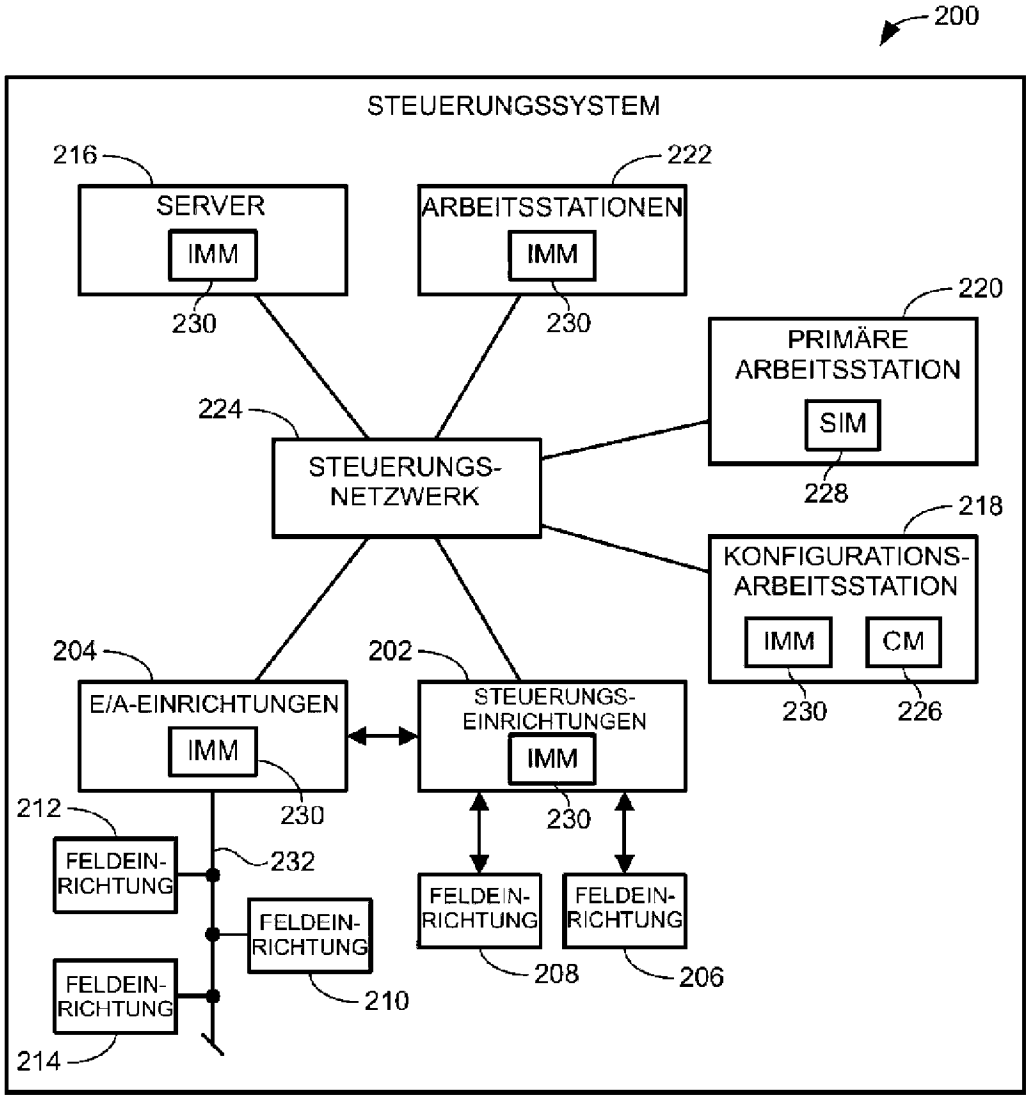

200
STEUERUNGSSYSTEM
216
SERVER
IMM
230
222
ARBEITSSTATIONEN
IMM
230
220
PRIMÄRE ARBEITSSTATION
SIM
228
224
STEUERUNGS-NETZWERK
218
KONFIGURATIONS-ARBEITSSTATION
IMM
CM
230
226
204
E/A-EINRICHTUNGEN
IMM
230
202
STEUERUNGS-EINRICHTUNGEN
IMM
230
212
FELDEIN-RICHTUNG
232
FELDEIN-RICHTUNG
FELDEIN-RICHTUNG
210
214
FELDEIN-RICHTUNG
208
FELDEIN-RICHTUNG
206

FIG. 2

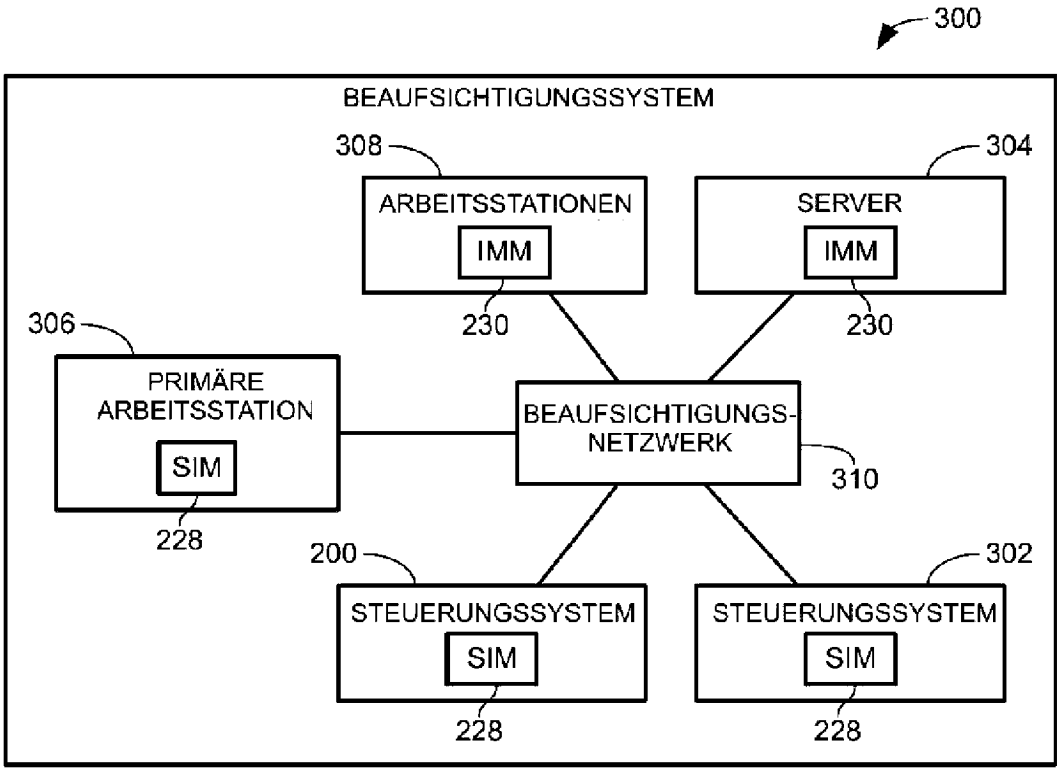

300
BEAUFSICHTIGUNGSSYSTEM
308
ARBEITSSTATIONEN
IMM
230
304
SERVER
IMM
230
306
PRIMÄRE ARBEITSSTATION
SIM
228
BEAUFSICHTIGUNGS-NETZWERK
310
200
STEUERUNGSSYSTEM
SIM
228
302
STEUERUNGSSYSTEM
SIM
228

FIG. 3

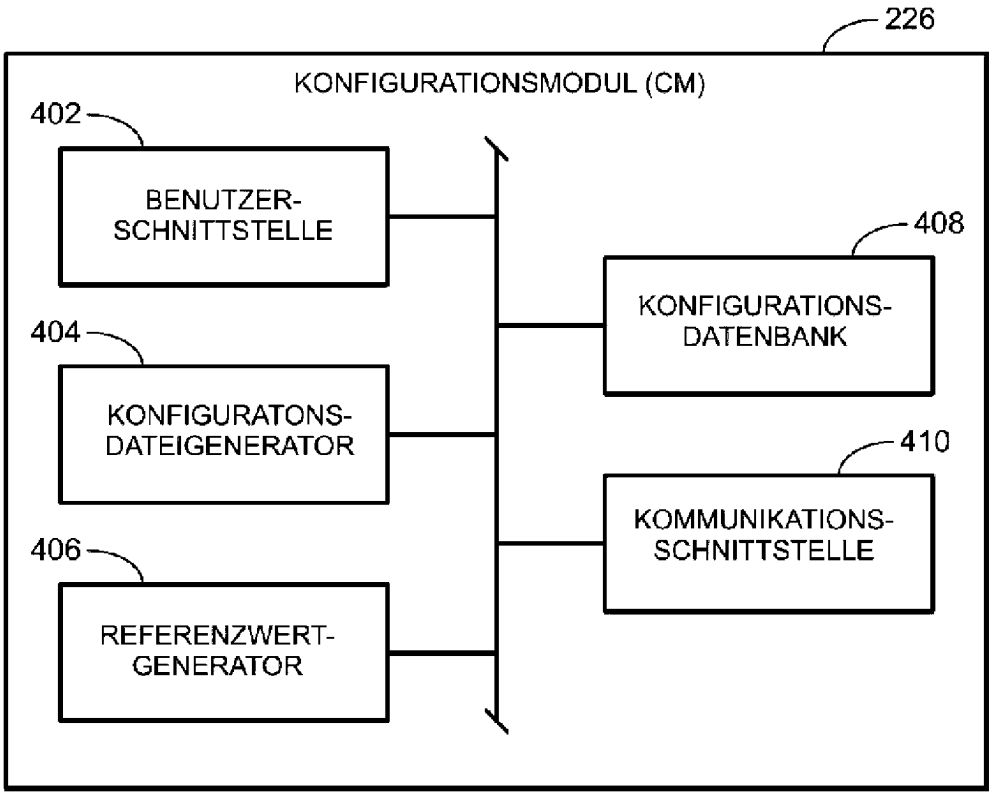

226
KONFIGURATIONSMODUL (CM)
402
BENUTZER-SCHNITTSTELLE
404
KONFIGURATONS-DATEIGENERATOR
406
REFERENZWERT-GENERATOR
408
KONFIGURATIONS-DATENBANK
410
KOMMUNIKATIONS-SCHNITTSTELLE

FIG. 4

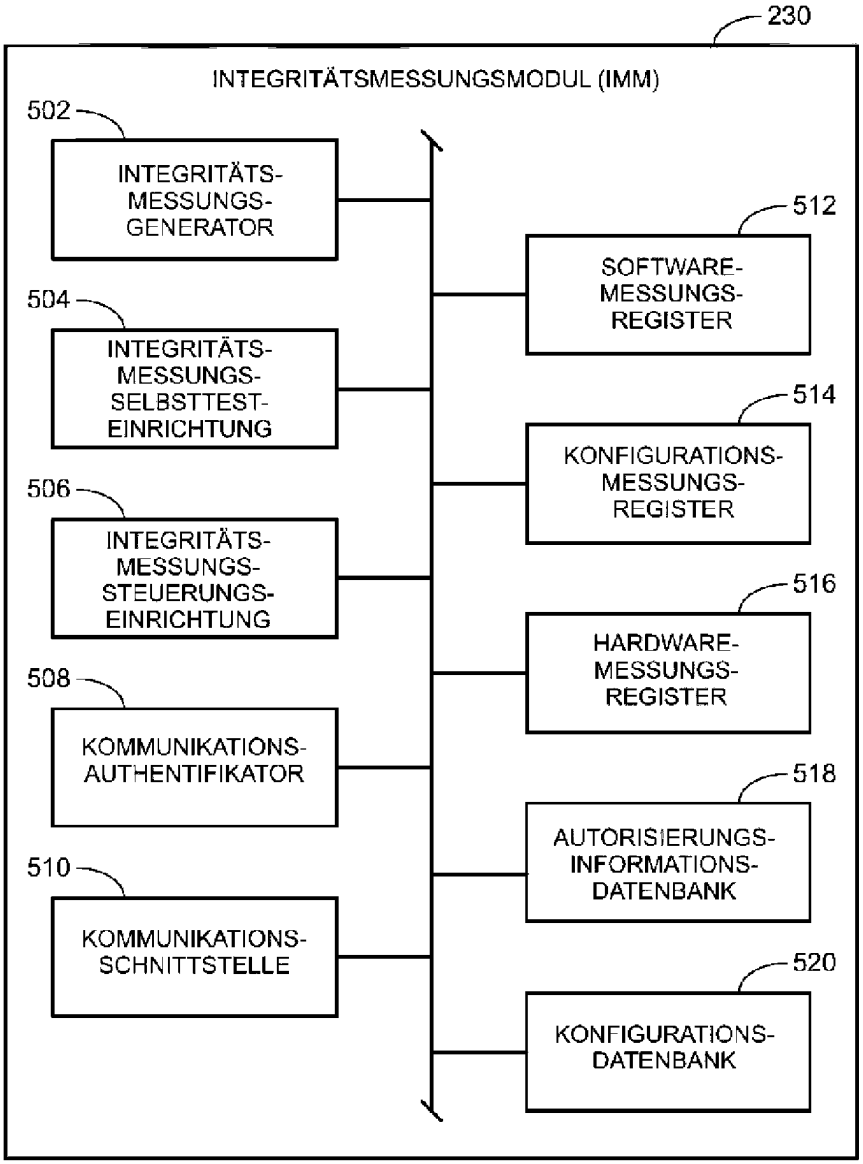

230
INTEGRITÄTSMESSUNGSMODUL (IMM)
502
INTEGRITÄTS-MESSUNGS-GENERATOR
504
INTEGRITÄTS-MESSUNGS-SELBSTTEST-EINRICHTUNG
506
INTEGRITÄTS-MESSUNGS-STEUERUNGS-EINRICHTUNG
508
KOMMUNIKATIONS-AUTHENTIFIKATOR
510
KOMMUNIKATIONS-SCHNITTSTELLE
512
SOFTWARE-MESSUNGS-REGISTER
514
KONFIGURATIONS-MESSUNGS-REGISTER
516
HARDWARE-MESSUNGS-REGISTER
518
AUTORISIERUNGS-INFORMATIONS-DATENBANK
520
KONFIGURATIONS-DATENBANK

FIG. 5

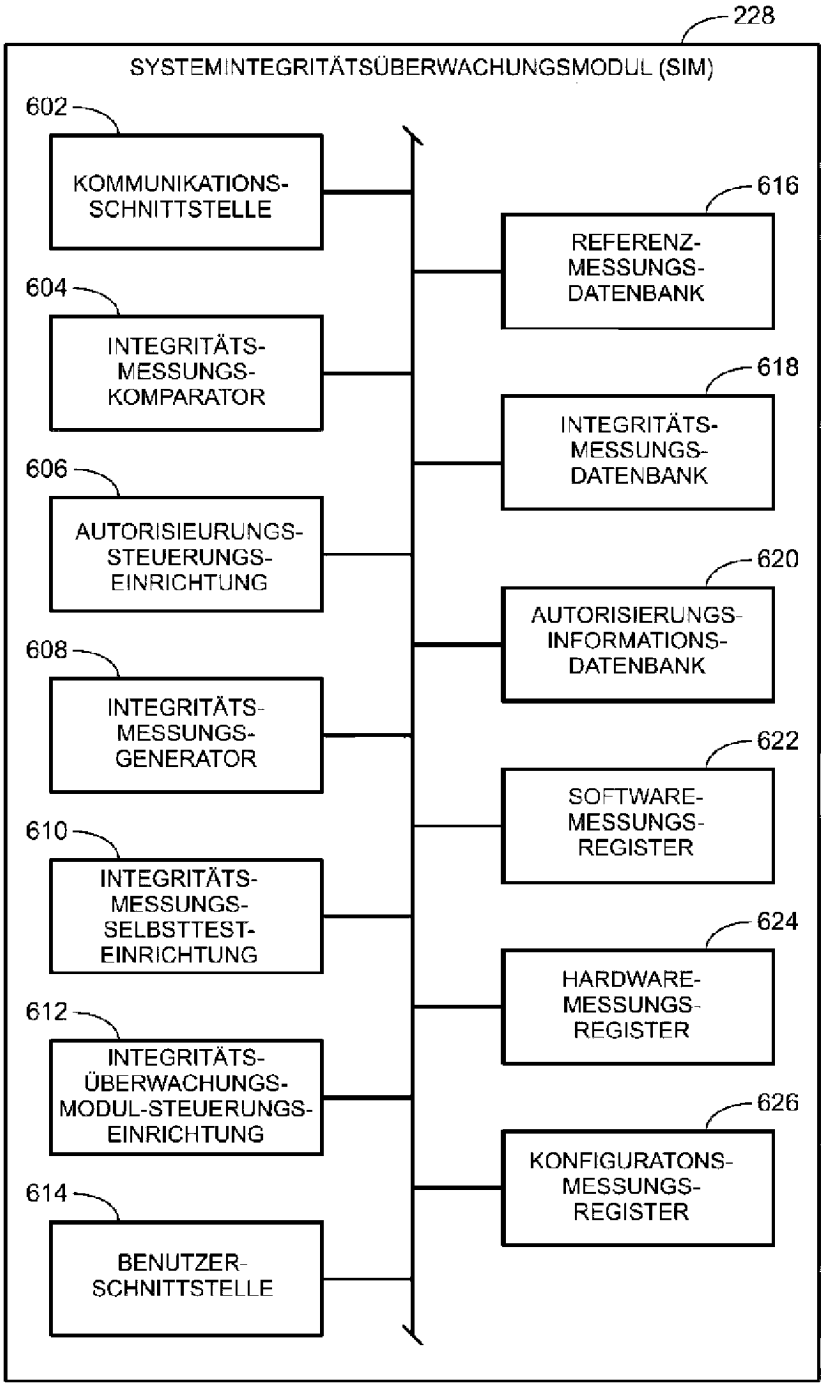

228
SYSTEMINTEGRITÄTSÜBERWACHUNGSMODUL (SIM)
602
KOMMUNIKATIONS-SCHNITTSTELLE
604
INTEGRITÄTS-MESSUNGS-KOMPARATOR
606
AUTORISIEURUNGS-STEUERUNGS-EINRICHTUNG
608
INTEGRITÄTS-MESSUNGS-GENERATOR
610
INTEGRITÄTS-MESSUNGS-SELBSTTEST-EINRICHTUNG
612
INTEGRITÄTS-ÜBERWACHUNGS-MODUL-STEUERUNGS-EINRICHTUNG
614
BENUTZER-SCHNITTSTELLE
616
REFERENZ-MESSUNGS-DATENBANK
618
INTEGRITÄTS-MESSUNGS-DATENBANK
620
AUTORISIERUNGS-INFORMATIONS-DATENBANK
622
SOFTWARE-MESSUNGS-REGISTER
624
HARDWARE-MESSUNGS-REGISTER
626
KONFIGURATONS-MESSUNGS-REGISTER

FIG. 6

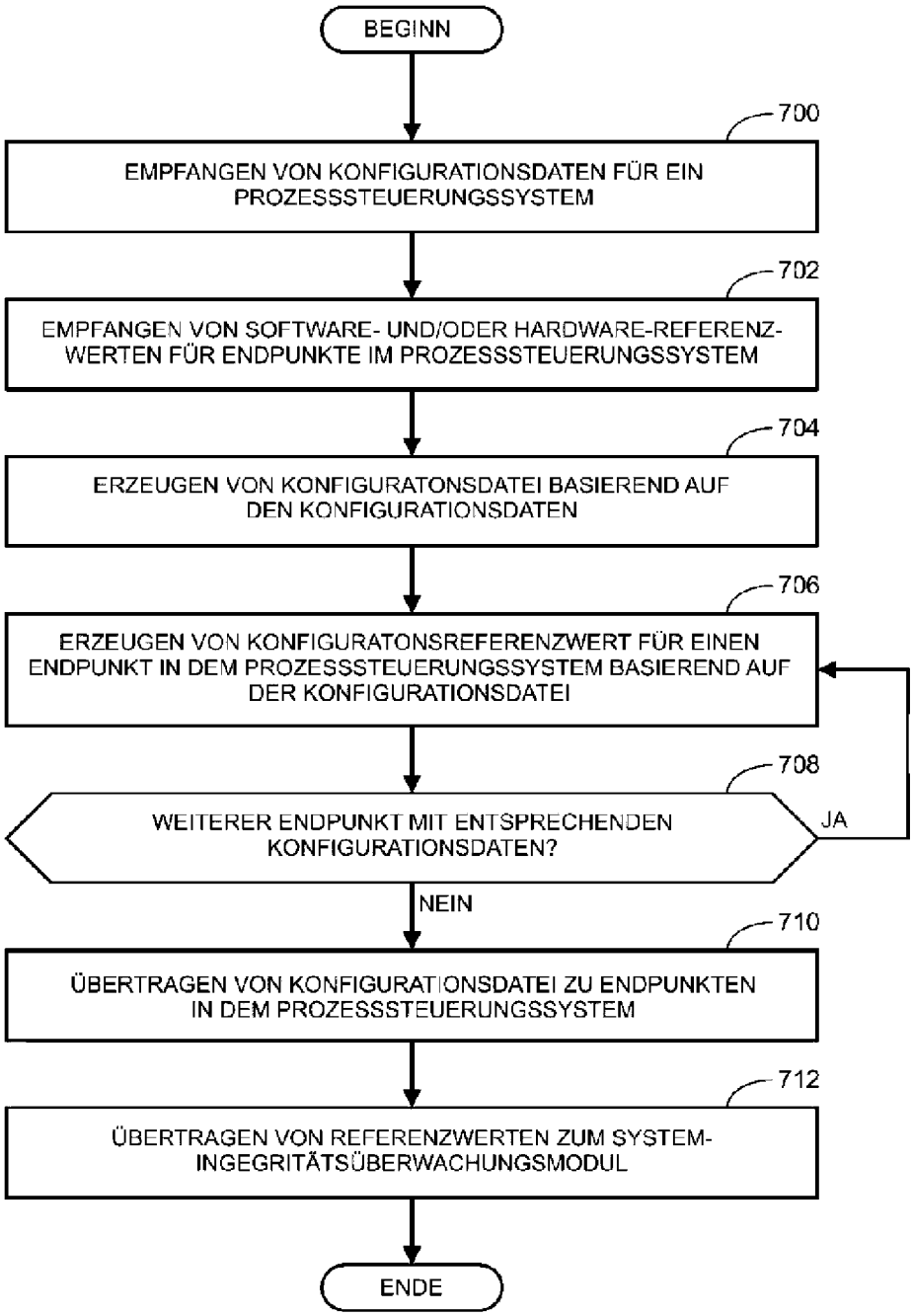

BEGINN
700
EMPFANGEN VON KONFIGURATIONSDATEN FÜR EIN PROZESSSTEUERUNGSSYSTEM
702
EMPFANGEN VON SOFTWARE- UND/ODER HARDWARE-REFERENZ-WERTEN FÜR ENDPUNKTE IM PROZESSSTEUERUNGSSYSTEM
704
ERZEUGEN VON KONFIGURATONSDATEI BASIEREND AUF DEN KONFIGURATIONSDATEN
706
ERZEUGEN VON KONFIGURATONSREFERENZWERT FÜR EINEN ENDPUNKT IN DEM PROZESSSTEUERUNGSSYSTEM BASIEREND AUF DER KONFIGURATIONSDATEI
708
WEITERER ENDPUNKT MIT ENTSPRECHENDEN KONFIGURATIONSDATEN?
JA
NEIN
710
ÜBERTRAGEN VON KONFIGURATIONSDATEI ZU ENDPUNKTEN IN DEM PROZESSSTEUERUNGSSYSTEM
712
ÜBERTRAGEN VON REFERENZWERTEN ZUM SYSTEM-INGEGRITÄTSÜBERWACHUNGSMODUL
ENDE

FIG. 7

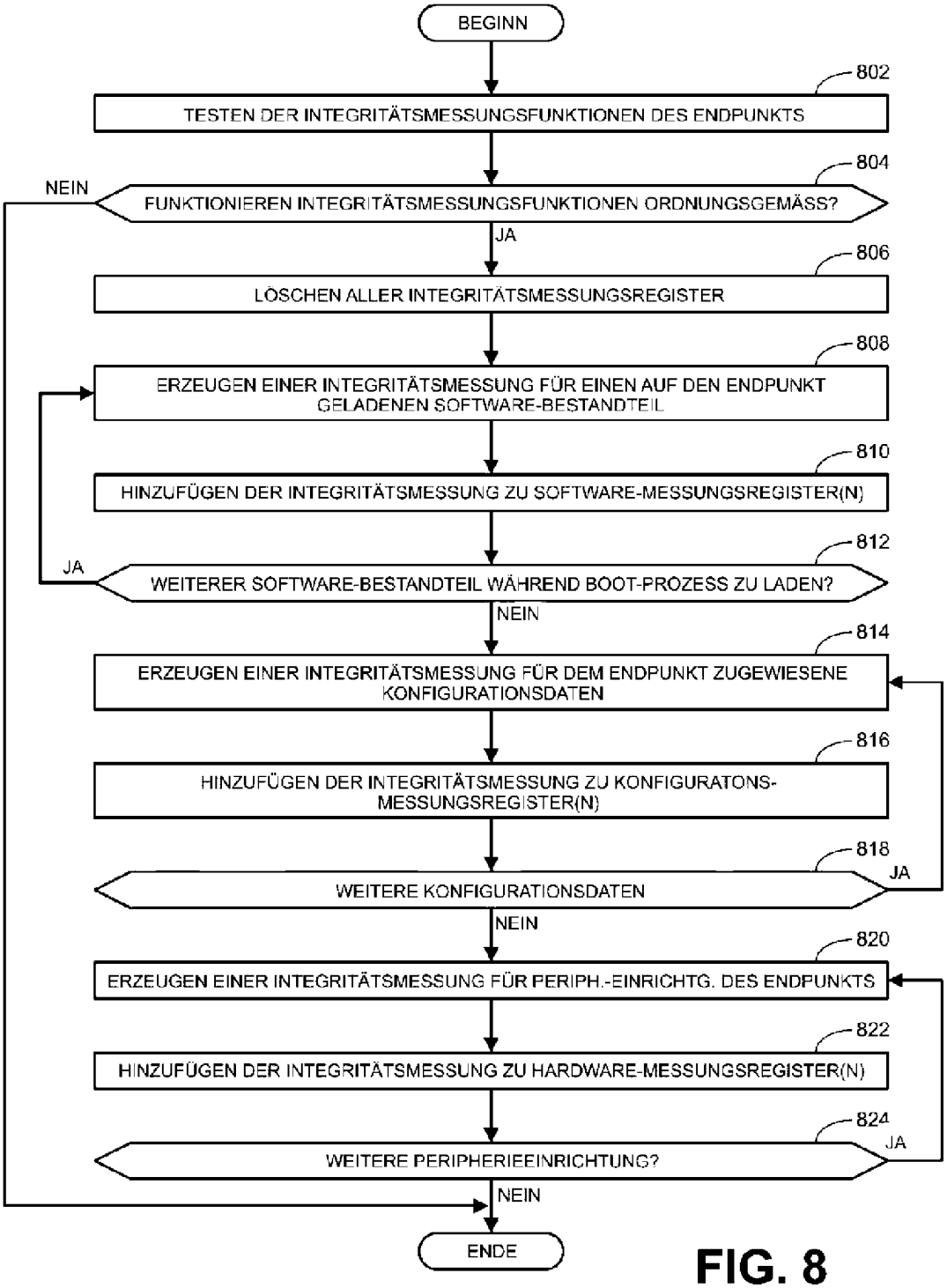

BEGINN
802
TESTEN DER INTEGRITÄTSMESSUNGSFUNKTIONEN DES ENDPUNKTS
804
NEIN
FUNKTIONIEREN INTEGRITÄTSMESSUNGSFUNKTIONEN ORDNUNGSGEMÄSS?
JA
806
LÖSCHEN ALLER INTEGRITÄTSMESSUNGSREGISTER
808
ERZEUGEN EINER INTEGRITÄTSMESSUNG FÜR EINEN AUF DEN ENDPUNKT GELADENEN SOFTWARE-BESTANDTEIL
810
HINZUFÜGEN DER INTEGRITÄTSMESSUNG ZU SOFTWARE-MESSUNGSREGISTER(N)
812
JA
WEITERER SOFTWARE-BESTANDTEIL WÄHREND BOOT-PROZESS ZU LADEN?
NEIN
814
ERZEUGEN EINER INTEGRITÄTSMESSUNG FÜR DEM ENDPUNKT ZUGEWIESENE KONFIGURATIONSDATEN
816
HINZUFÜGEN DER INTEGRITÄTSMESSUNG ZU KONFIGURATONS-MESSUNGSREGISTER(N)
818
JA
WEITERE KONFIGURATIONSDATEN
NEIN
820
ERZEUGEN EINER INTEGRITÄTSMESSUNG FÜR PERIPH.-EINRICHTG. DES ENDPUNKTS
822
HINZUFÜGEN DER INTEGRITÄTSMESSUNG ZU HARDWARE-MESSUNGSREGISTER(N)
824
JA
WEITERE PERIPHERIEEINRICHTUNG?
NEIN
ENDE

FIG. 8

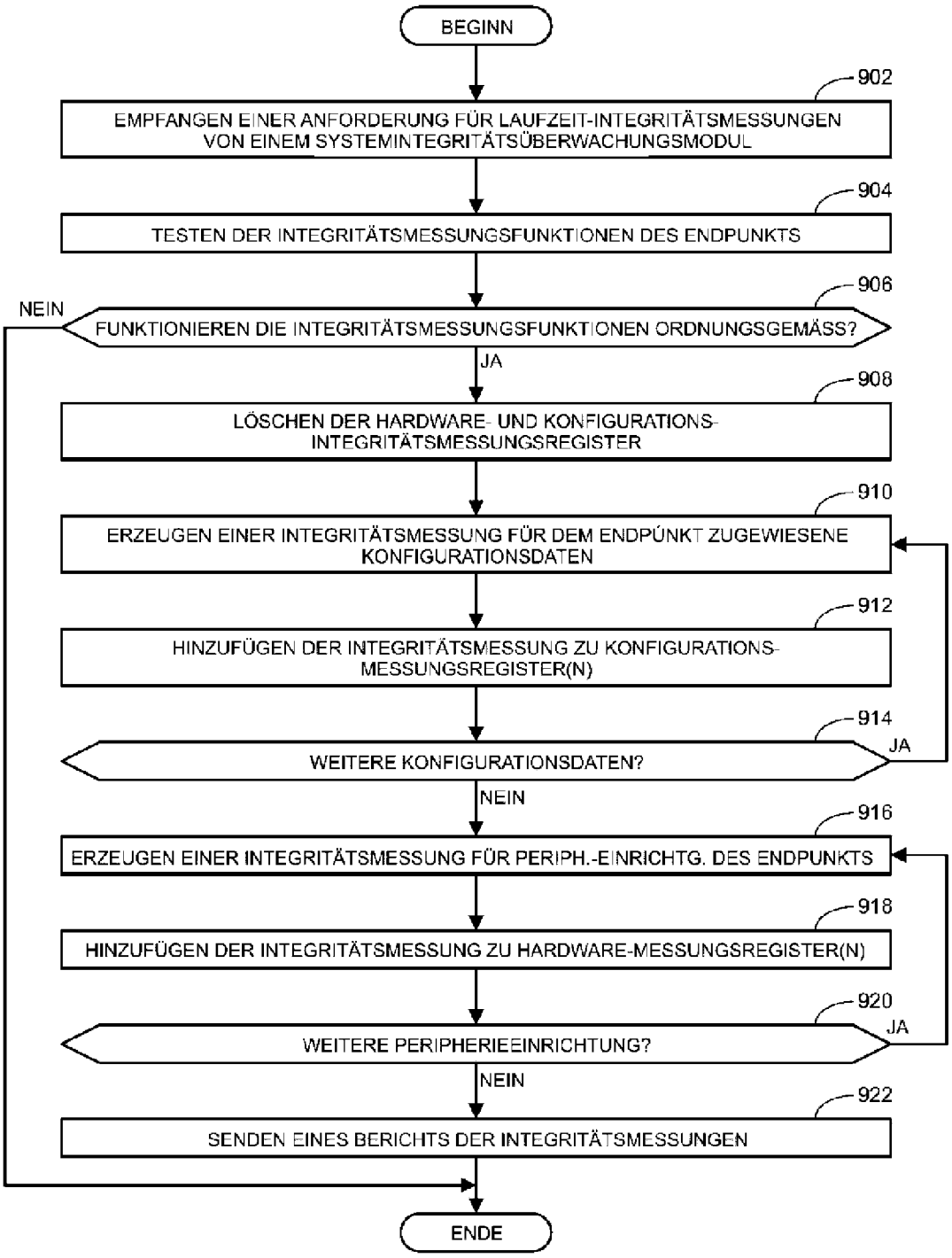

BEGINN
902
EMPFANGEN EINER ANFORDERUNG FÜR LAUFZEIT-INTEGRITÄTSMESSUNGEN VON EINEM SYSTEMINTEGRITÄTSÜBERWACHUNGSMODUL
904
TESTEN DER INTEGRITÄTSMESSUNGSFUNKTIONEN DES ENDPUNKTS
906
NEIN
FUNKTIONIEREN DIE INTEGRITÄTSMESSUNGSFUNKTIONEN ORDNUNGSGEMÄSS?
JA
908
LÖSCHEN DER HARDWARE- UND KONFIGURATIONS-INTEGRITÄTSMESSUNGSREGISTER
910
ERZEUGEN EINER INTEGRITÄTSMESSUNG FÜR DEM ENDPUNKT ZUGEWIESENE KONFIGURATIONSDATEN
912
HINZUFÜGEN DER INTEGRITÄTSMESSUNG ZU KONFIGURATIONS-MESSUNGSREGISTER(N)
914
JA
WEITERE KONFIGURATIONSDATEN?
NEIN
916
ERZEUGEN EINER INTEGRITÄTSMESSUNG FÜR PERIPH.-EINRICHTG. DES ENDPUNKTS
918
HINZUFÜGEN DER INTEGRITÄTSMESSUNG ZU HARDWARE-MESSUNGSREGISTER(N)
920
JA
WEITERE PERIPHERIEEINRICHTUNG?
NEIN
922
SENDEN EINES BERICHTS DER INTEGRITÄTSMESSUNGEN
ENDE

FIG. 9

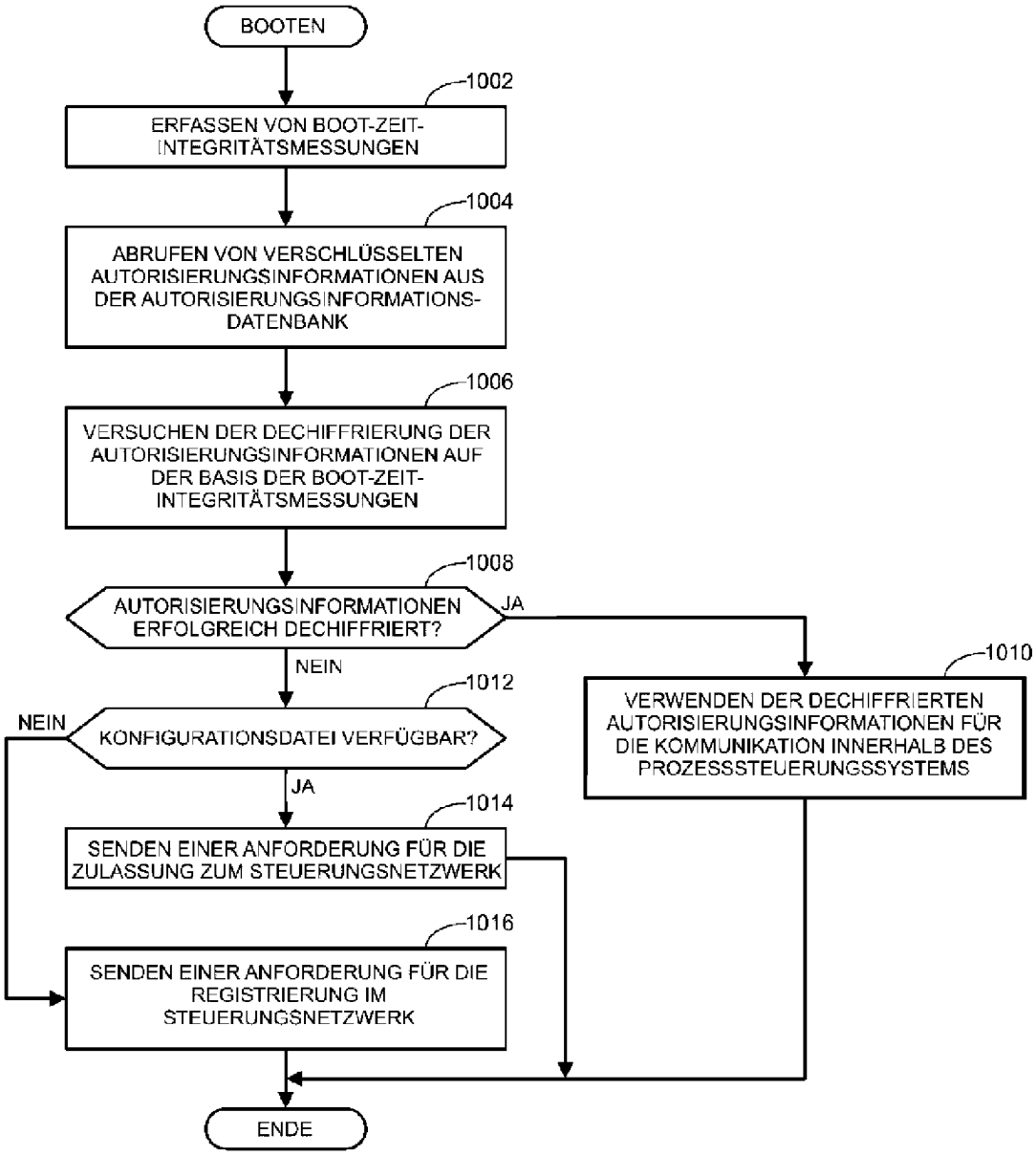

BOOTEN
1002
ERFASSEN VON BOOT-ZEIT-INTEGRITÄTSMESSUNGEN
1004
ABRUFEN VON VERSCHLÜSSELTEN AUTORISIERUNGSINFORMATIONEN AUS DER AUTORISIERUNGSINFORMATIONS-DATENBANK
1006
VERSUCHEN DER DECHIFFRIERUNG DER AUTORISIERUNGSINFORMATIONEN AUF DER BASIS DER BOOT-ZEIT-INTEGRITÄTSMESSUNGEN
1008
AUTORISIERUNGSINFORMATIONEN ERFOLGREICH DECHIFFRIERT?
JA
NEIN
1010
VERWENDEN DER DECHIFFRIERTEN AUTORISIERUNGSINFORMATIONEN FÜR DIE KOMMUNIKATION INNERHALB DES PROZESSSTEUERUNGSSYSTEMS
1012
NEIN
KONFIGURATIONSDATEI VERFÜGBAR?
JA
1014
SENDEN EINER ANFORDERUNG FÜR DIE ZULASSUNG ZUM STEUERUNGSNETZWERK
1016
SENDEN EINER ANFORDERUNG FÜR DIE REGISTRIERUNG IM STEUERUNGSNETZWERK
ENDE

FIG. 10

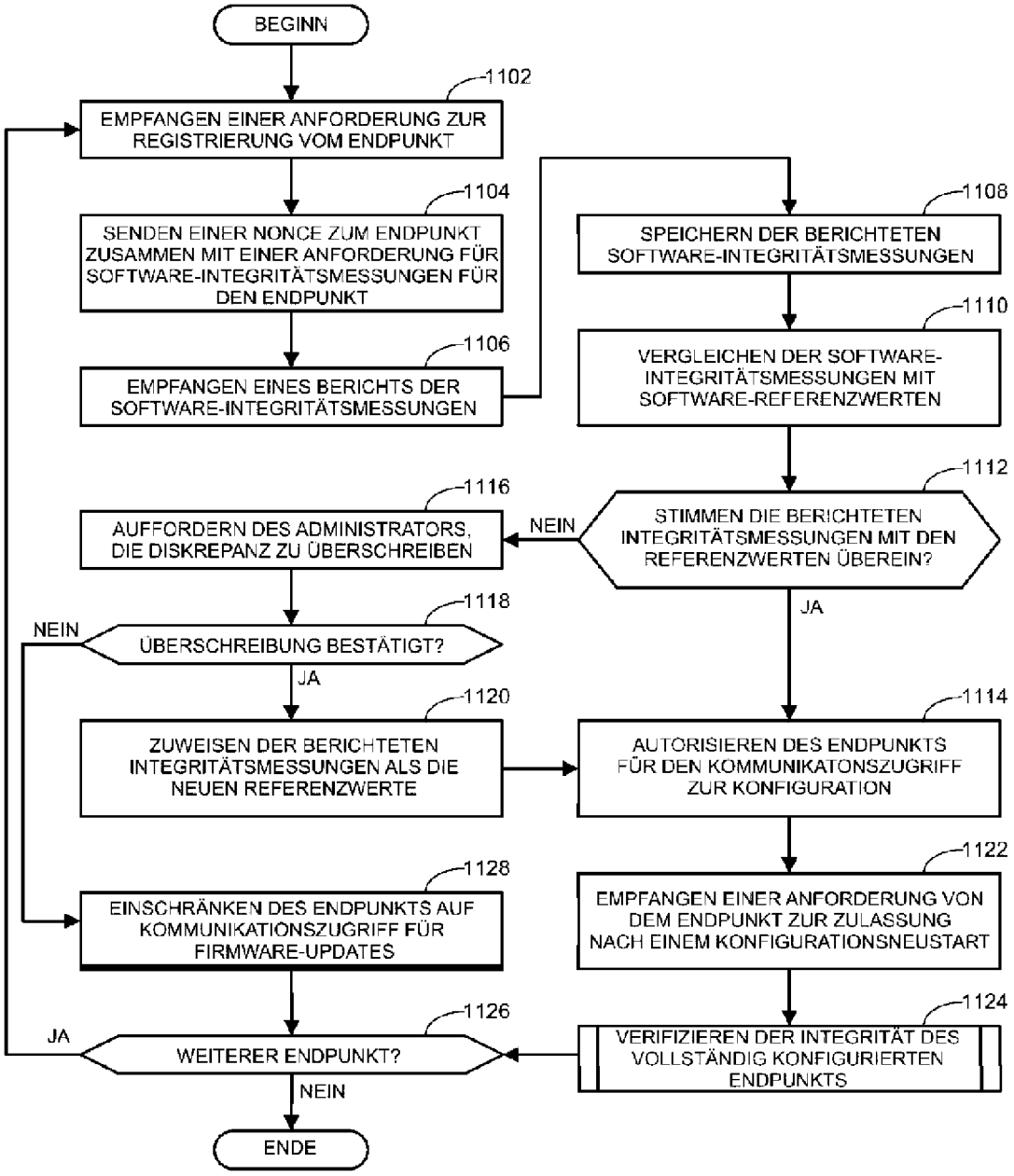

BEGINN
1102
EMPFANGEN EINER ANFORDERUNG ZUR REGISTRIERUNG VOM ENDPUNKT
1104
SENDEN EINER NONCE ZUM ENDPUNKT ZUSAMMEN MIT EINER ANFORDERUNG FÜR SOFTWARE-INTEGRITÄTSMESSUNGEN FÜR DEN ENDPUNKT
1106
EMPFANGEN EINES BERICHTS DER SOFTWARE-INTEGRITÄTSMESSUNGEN
1108
SPEICHERN DER BERICHTETEN SOFTWARE-INTEGRITÄTSMESSUNGEN
1110
VERGLEICHEN DER SOFTWARE-INTEGRITÄTSMESSUNGEN MIT SOFTWARE-REFERENZWERTEN
1112
STIMMEN DIE BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN ÜBEREIN?
NEIN
JA
1116
AUFFORDERN DES ADMINISTRATORS, DIE DISKREPANZ ZU ÜBERSCHREIBEN
1118
ÜBERSCHREIBUNG BESTÄTIGT?
NEIN
JA
1120
ZUWEISEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN ALS DIE NEUEN REFERENZWERTE
1114
AUTORISIEREN DES ENDPUNKTS FÜR DEN KOMMUNIKATONSZUGRIFF ZUR KONFIGURATION
1122
EMPFANGEN EINER ANFORDERUNG VON DEM ENDPUNKT ZUR ZULASSUNG NACH EINEM KONFIGURATIONSNEUSTART
1124
VERIFIZIEREN DER INTEGRITÄT DES VOLLSTÄNDIG KONFIGURIERTEN ENDPUNKTS
1128
EINSCHRÄNKEN DES ENDPUNKTS AUF KOMMUNIKATIONSZUGRIFF FÜR FIRMWARE-UPDATES
1126
JA
WEITERER ENDPUNKT?
NEIN
ENDE

FIG. 11

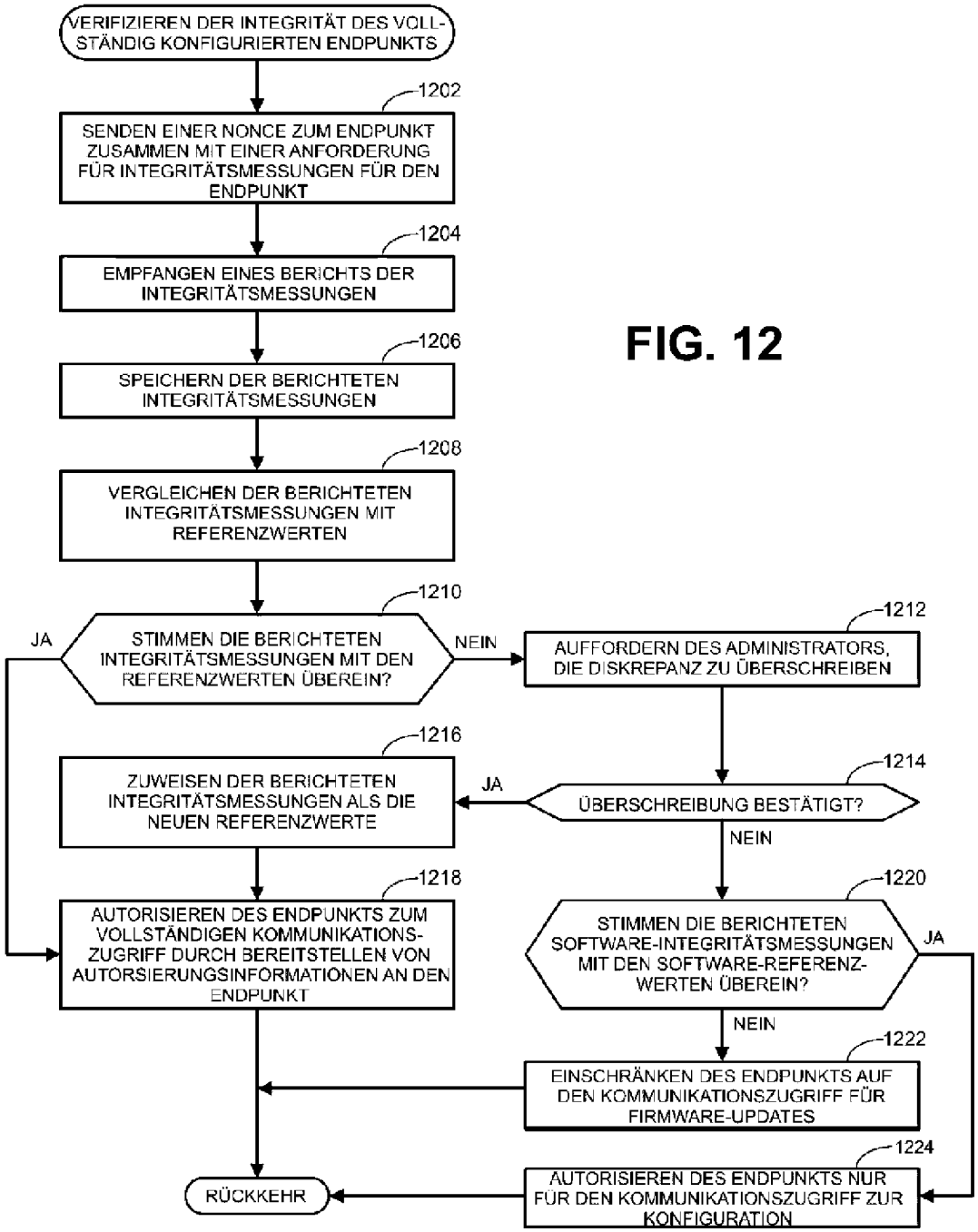

VERIFIZIEREN DER INTEGRITÄT DES VOLLSTÄNDIG KONFIGURIERTEN ENDPUNKTS
1202
SENDEN EINER NONCE ZUM ENDPUNKT ZUSAMMEN MIT EINER ANFORDERUNG FÜR INTEGRITÄTSMESSUNGEN FÜR DEN ENDPUNKT
1204
EMPFANGEN EINES BERICHTS DER INTEGRITÄTSMESSUNGEN
1206
SPEICHERN DER BERICHTETEN INTEGRITÄTSMESSUNGEN
1208
VERGLEICHEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN MIT REFERENZWERTEN
1210
JA
STIMMEN DIE BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN ÜBEREIN?
NEIN
1212
AUFFORDERN DES ADMINISTRATORS, DIE DISKREPANZ ZU ÜBERSCHREIBEN
1214
ÜBERSCHREIBUNG BESTÄTIGT?
JA
1216
ZUWEISEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN ALS DIE NEUEN REFERENZWERTE
NEIN
1220
STIMMEN DIE BERICHTETEN SOFTWARE-INTEGRITÄTSMESSUNGEN MIT DEN SOFTWARE-REFERENZ-WERTEN ÜBEREIN?
JA
NEIN
1218
AUTORISIEREN DES ENDPUNKTS ZUM VOLLSTÄNDIGEN KOMMUNIKATIONS-ZUGRIFF DURCH BEREITSTELLEN VON AUTORSIERUNGSINFORMATIONEN AN DEN ENDPUNKT
1222
EINSCHRÄNKEN DES ENDPUNKTS AUF DEN KOMMUNIKATIONSZUGRIFF FÜR FIRMWARE-UPDATES
1224
AUTORISIEREN DES ENDPUNKTS NUR FÜR DEN KOMMUNIKATIONSZUGRIFF ZUR KONFIGURATION
RÜCKKEHR

FIG. 12

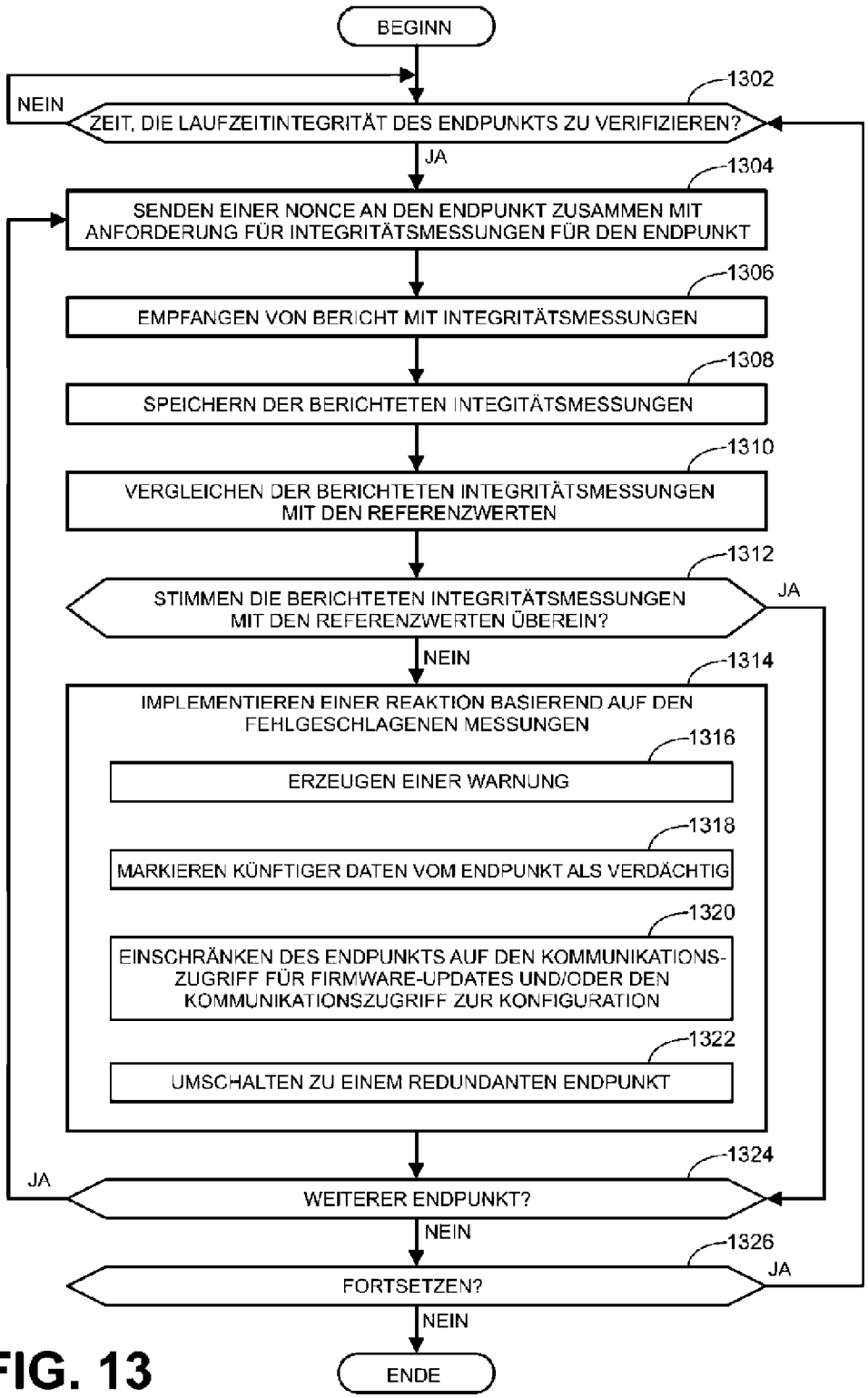

BEGINN
1302
NEIN
ZEIT, DIE LAUFZEITINTEGRITÄT DES ENDPUNKTS ZU VERIFIZIEREN?
JA
1304
SENDEN EINER NONCE AN DEN ENDPUNKT ZUSAMMEN MIT ANFORDERUNG FÜR INTEGRITÄTSMESSUNGEN FÜR DEN ENDPUNKT
1306
EMPFANGEN VON BERICHT MIT INTEGRITÄTSMESSUNGEN
1308
SPEICHERN DER BERICHTETEN INTEGITÄTSMESSUNGEN
1310
VERGLEICHEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN
1312
STIMMEN DIE BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN ÜBEREIN?
JA
NEIN
1314
IMPLEMENTIEREN EINER REAKTION BASIEREND AUF DEN FEHLGESCHLAGENEN MESSUNGEN
1316
ERZEUGEN EINER WARNUNG
1318
MARKIEREN KÜNFTIGER DATEN VOM ENDPUNKT ALS VERDÄCHTIG
1320
EINSCHRÄNKEN DES ENDPUNKTS AUF DEN KOMMUNIKATIONS-ZUGRIFF FÜR FIRMWARE-UPDATES UND/ODER DEN KOMMUNIKATIONSZUGRIFF ZUR KONFIGURATION
1322
UMSCHALTEN ZU EINEM REDUNDANTEN ENDPUNKT
1324
JA
WEITERER ENDPUNKT?
NEIN
1326
FORTSETZEN?
JA
NEIN
ENDE

FIG. 13

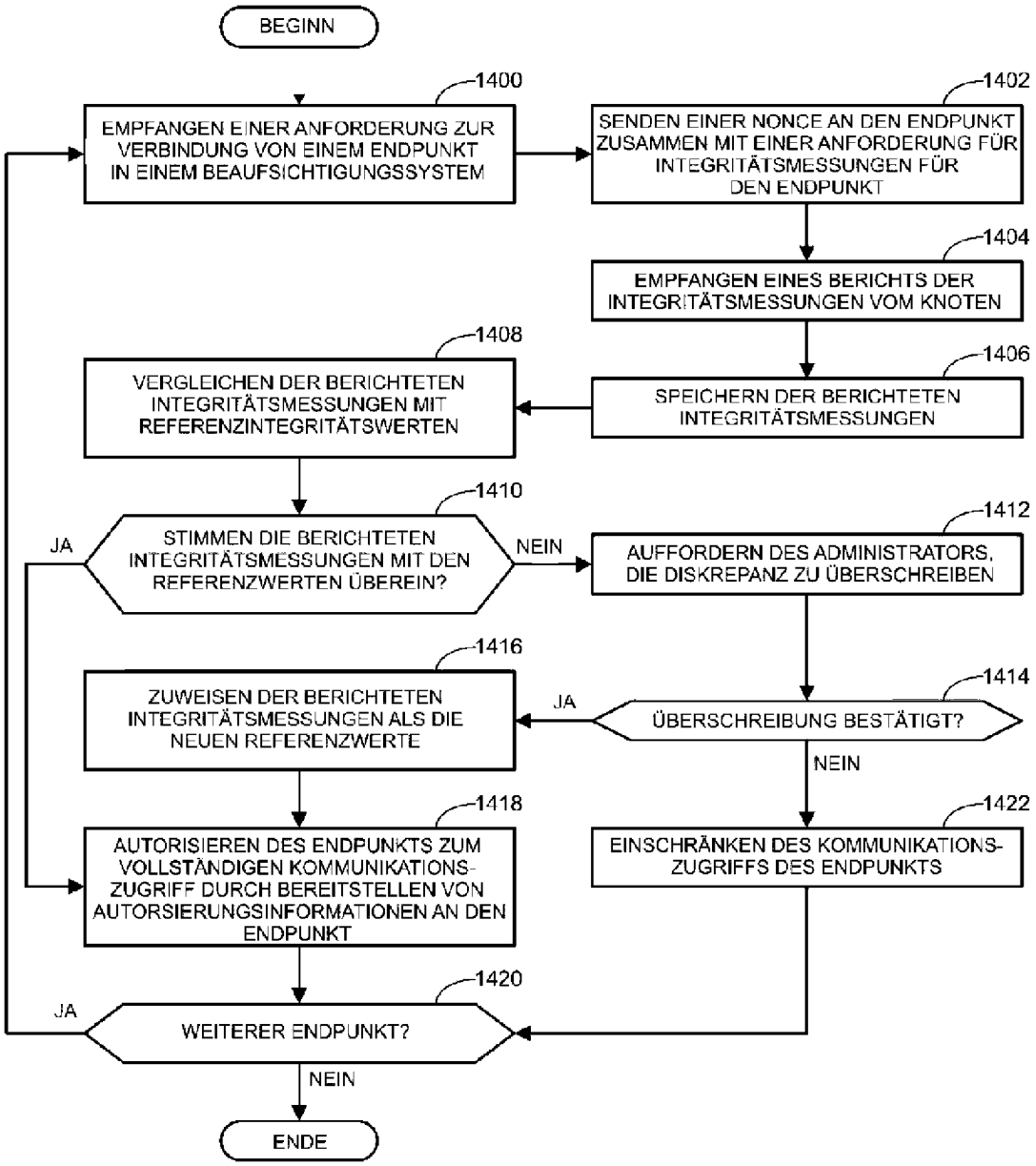

BEGINN
1400
EMPFANGEN EINER ANFORDERUNG ZUR VERBINDUNG VON EINEM ENDPUNKT IN EINEM BEAUFSICHTIGUNGSSYSTEM
1402
SENDEN EINER NONCE AN DEN ENDPUNKT ZUSAMMEN MIT EINER ANFORDERUNG FÜR INTEGRITÄTSMESSUNGEN FÜR DEN ENDPUNKT
1404
EMPFANGEN EINES BERICHTS DER INTEGRITÄTSMESSUNGEN VOM KNOTEN
1406
SPEICHERN DER BERICHTETEN INTEGRITÄTSMESSUNGEN
1408
VERGLEICHEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN MIT REFERENZINTEGRITÄTSWERTEN
1410
STIMMEN DIE BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN ÜBEREIN?
JA
NEIN
1412
AUFFORDERN DES ADMINISTRATORS, DIE DISKREPANZ ZU ÜBERSCHREIBEN
1414
ÜBERSCHREIBUNG BESTÄTIGT?
JA
NEIN
1416
ZUWEISEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN ALS DIE NEUEN REFERENZWERTE
1418
AUTORISIEREN DES ENDPUNKTS ZUM VOLLSTÄNDIGEN KOMMUNIKATIONS-ZUGRIFF DURCH BEREITSTELLEN VON AUTORSIERUNGSINFORMATIONEN AN DEN ENDPUNKT
1422
EINSCHRÄNKEN DES KOMMUNIKATIONS-ZUGRIFFS DES ENDPUNKTS
1420
WEITERER ENDPUNKT?
JA
NEIN
ENDE

FIG. 14

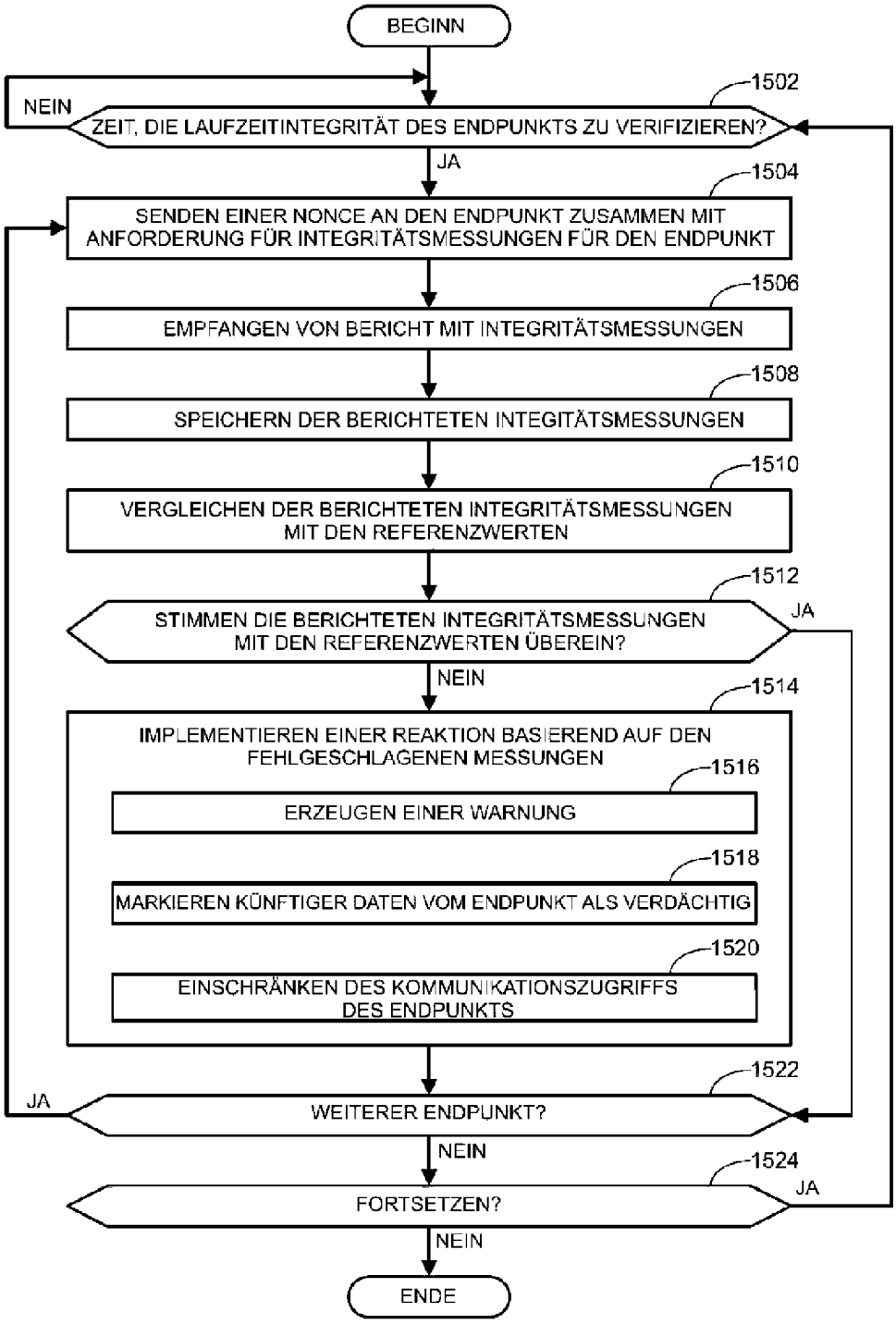

BEGINN
1502
NEIN
ZEIT, DIE LAUFZEITINTEGRITÄT DES ENDPUNKTS ZU VERIFIZIEREN?
JA
1504
SENDEN EINER NONCE AN DEN ENDPUNKT ZUSAMMEN MIT ANFORDERUNG FÜR INTEGRITÄTSMESSUNGEN FÜR DEN ENDPUNKT
1506
EMPFANGEN VON BERICHT MIT INTEGRITÄTSMESSUNGEN
1508
SPEICHERN DER BERICHTETEN INTEGITÄTSMESSUNGEN
1510
VERGLEICHEN DER BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN
1512
STIMMEN DIE BERICHTETEN INTEGRITÄTSMESSUNGEN MIT DEN REFERENZWERTEN ÜBEREIN?
JA
NEIN
1514
IMPLEMENTIEREN EINER REAKTION BASIEREND AUF DEN FEHLGESCHLAGENEN MESSUNGEN
1516
ERZEUGEN EINER WARNUNG
1518
MARKIEREN KÜNFTIGER DATEN VOM ENDPUNKT ALS VERDÄCHTIG
1520
EINSCHRÄNKEN DES KOMMUNIKATIONSZUGRIFFS DES ENDPUNKTS
1522
JA
WEITERER ENDPUNKT?
NEIN
1524
JA
FORTSETZEN?
NEIN
ENDE

FIG. 15

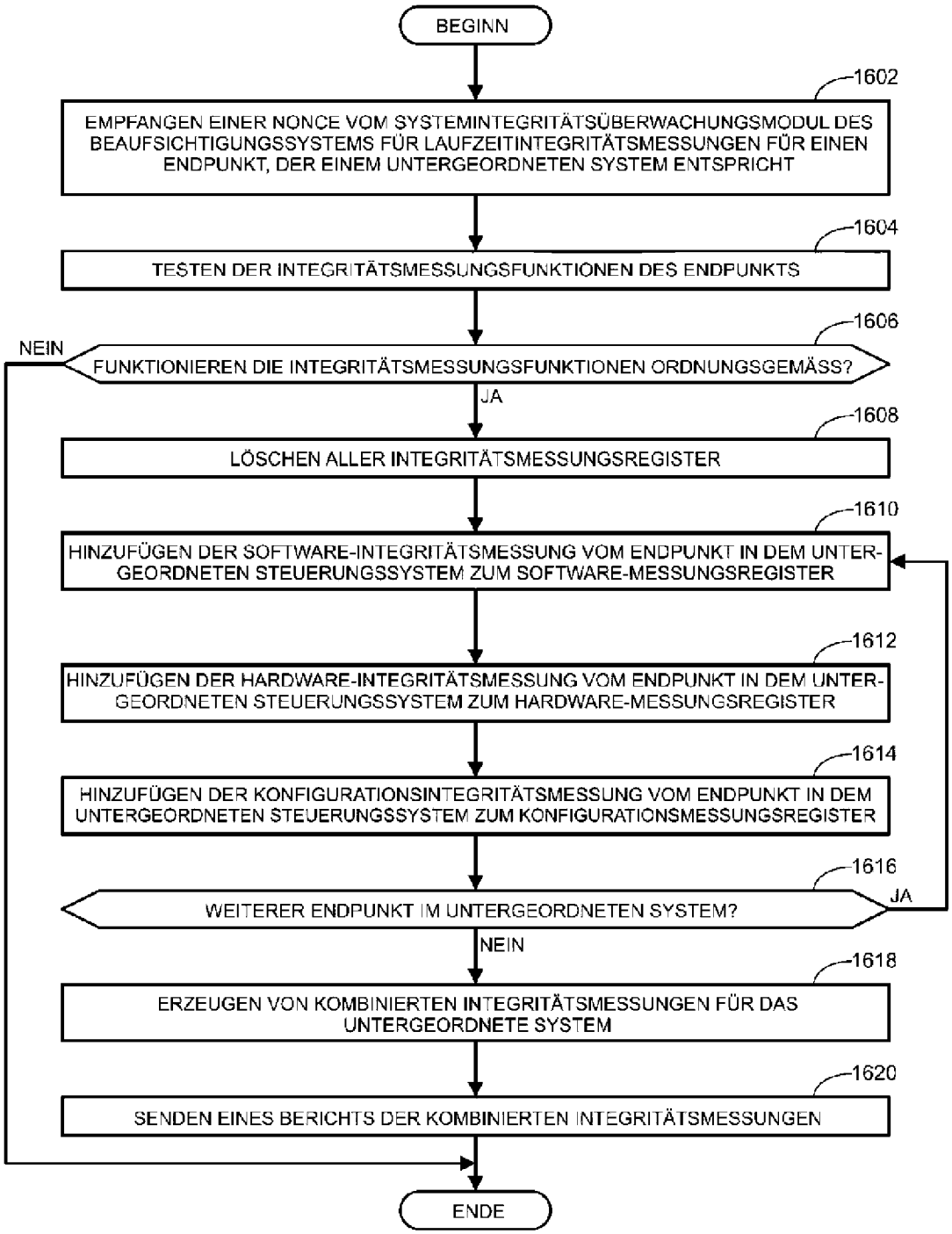

BEGINN
1602
EMPFANGEN EINER NONCE VOM SYSTEMINTEGRITÄTSÜBERWACHUNGSMODUL DES BEAUFSICHTIGUNGSSYSTEMS FÜR LAUFZEITINTEGRITÄTSMESSUNGEN FÜR EINEN ENDPUNKT, DER EINEM UNTERGEORDNETEN SYSTEM ENTSPRICHT
1604
TESTEN DER INTEGRITÄTSMESSUNGSFUNKTIONEN DES ENDPUNKTS
1606
NEIN
FUNKTIONIEREN DIE INTEGRITÄTSMESSUNGSFUNKTIONEN ORDNUNGSGEMÄSS?
JA
1608
LÖSCHEN ALLER INTEGRITÄTSMESSUNGSREGISTER
1610
HINZUFÜGEN DER SOFTWARE-INTEGRITÄTSMESSUNG VOM ENDPUNKT IN DEM UNTERGEORDNETEN STEUERUNGSSYSTEM ZUM SOFTWARE-MESSUNGSREGISTER
1612
HINZUFÜGEN DER HARDWARE-INTEGRITÄTSMESSUNG VOM ENDPUNKT IN DEM UNTERGEORDNETEN STEUERUNGSSYSTEM ZUM HARDWARE-MESSUNGSREGISTER
1614
HINZUFÜGEN DER KONFIGURATIONSINTEGRITÄTSMESSUNG VOM ENDPUNKT IN DEM UNTERGEORDNETEN STEUERUNGSSYSTEM ZUM KONFIGURATIONSMESSUNGSREGISTER
1616
JA
WEITERER ENDPUNKT IM UNTERGEORDNETEN SYSTEM?
NEIN
1618
ERZEUGEN VON KOMBINIERTEN INTEGRITÄTSMESSUNGEN FÜR DAS UNTERGEORDNETE SYSTEM
1620
SENDEN EINES BERICHTS DER KOMBINIERTEN INTEGRITÄTSMESSUNGEN
ENDE

FIG. 16

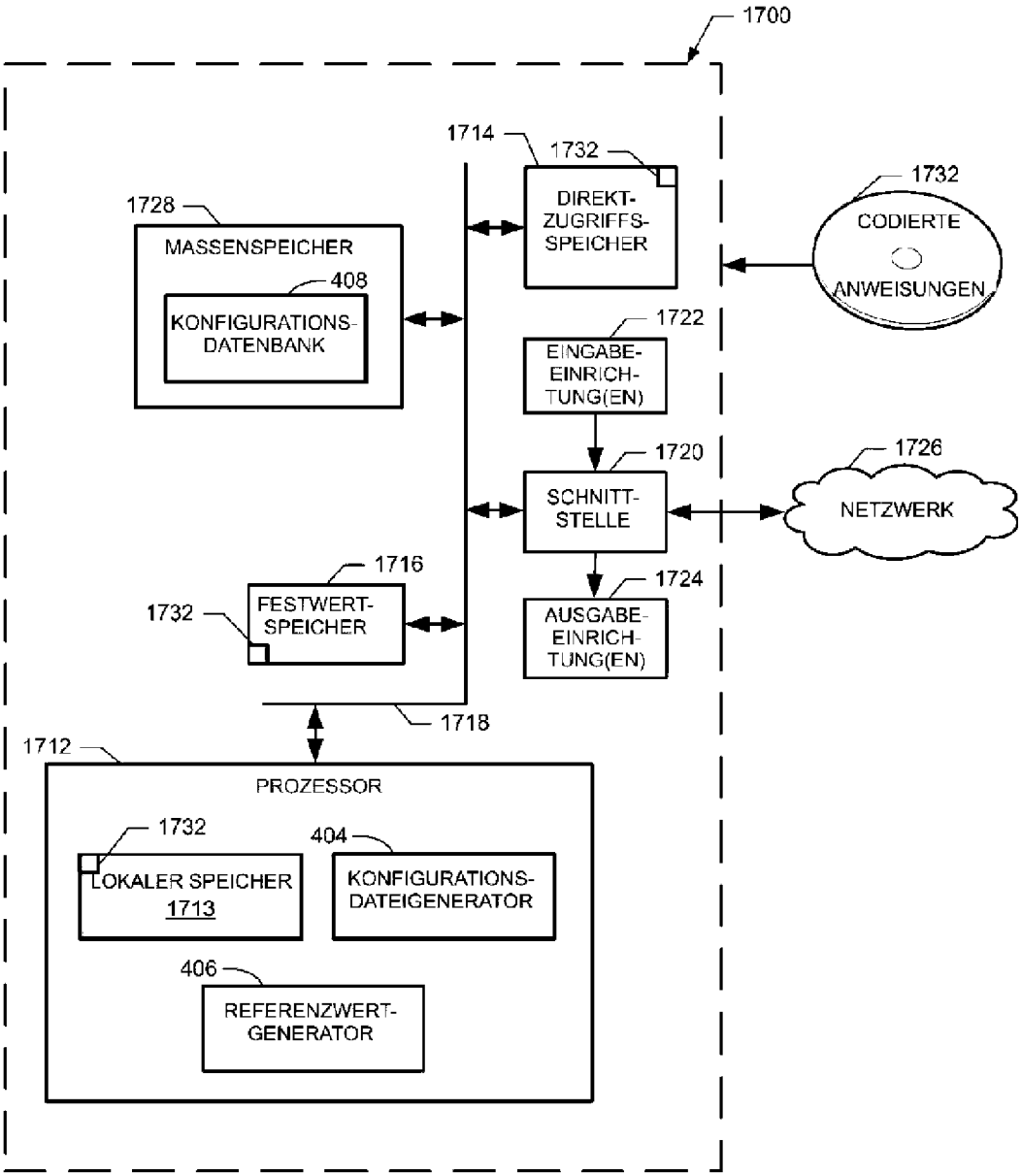

1700
1714
1732
DIREKT-ZUGRIFFS-SPEICHER
1732
CODIERTE ANWEISUNGEN
1728
MASSENSPEICHER
408
KONFIGURATIONS-DATENBANK
1722
EINGABE-EINRICH-TUNG(EN)
1720
SCHNITT-STELLE
1726
NETZWERK
1716
1732
FESTWERT-SPEICHER
1724
AUSGABE-EINRICH-TUNG(EN)
1718
1712
PROZESSOR
1732
LOKALER SPEICHER 1713
404
KONFIGURATIONS-DATEIGENERATOR
406
REFERENZWERT-GENERATOR

FIG. 17

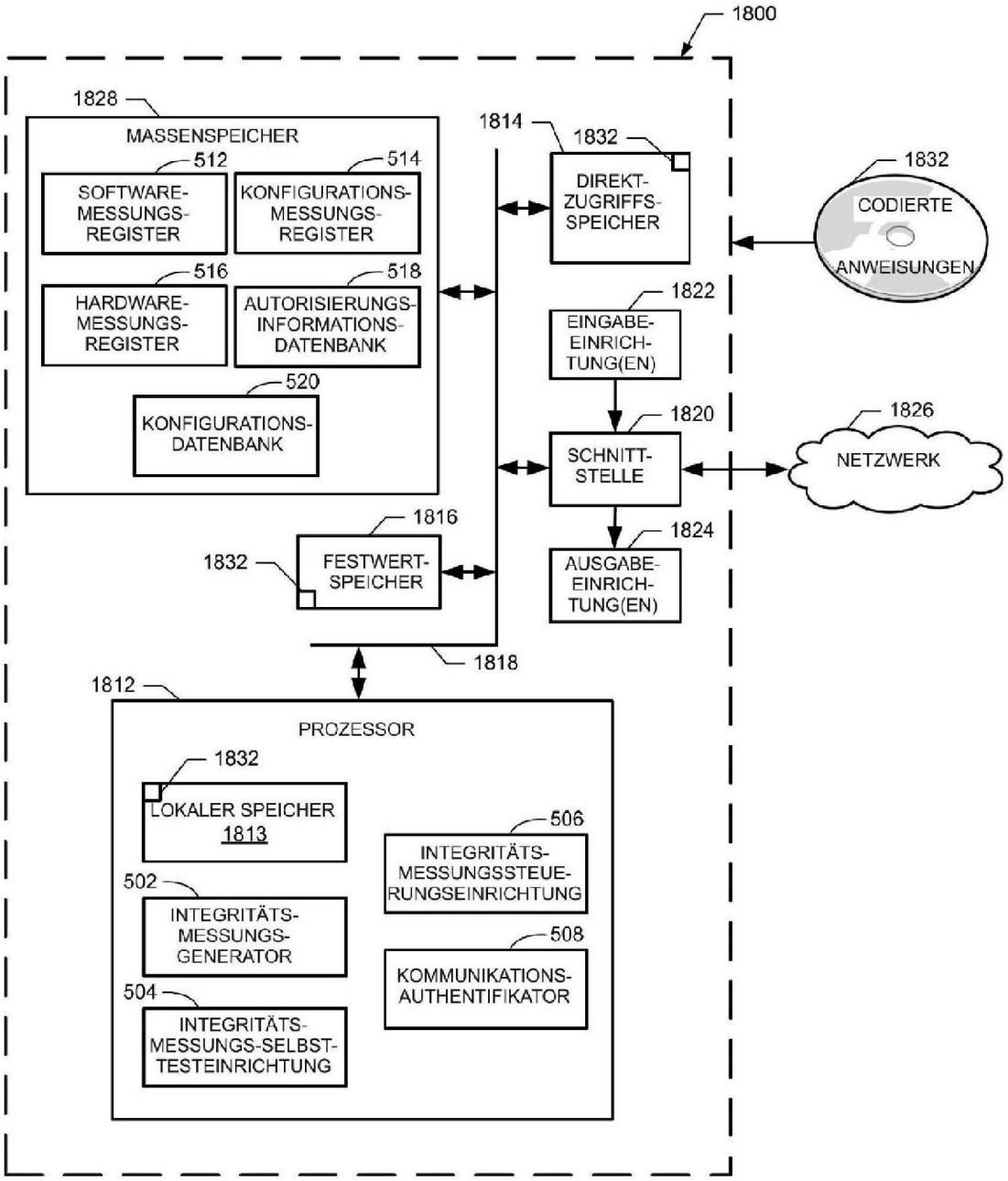

1800
1828
MASSENSPEICHER
512
SOFTWARE-MESSUNGS-REGISTER
514
KONFIGURATIONS-MESSUNGS-REGISTER
516
HARDWARE-MESSUNGS-REGISTER
518
AUTORISIERUNGS-INFORMATIONS-DATENBANK
520
KONFIGURATIONS-DATENBANK
1814
1832
DIREKT-ZUGRIFFS-SPEICHER
1832
CODIERTE ANWEISUNGEN
1822
EINGABE-EINRICH-TUNG(EN)
1820
SCHNITT-STELLE
1826
NETZWERK
1816
1832
FESTWERT-SPEICHER
1824
AUSGABE-EINRICH-TUNG(EN)
1818
1812
PROZESSOR
1832
LOKALER SPEICHER 1813
506
INTEGRITÄTS-MESSUNGSSTEUE-RUNGSEINRICHTUNG
502
INTEGRITÄTS-MESSUNGS-GENERATOR
508
KOMMUNIKATIONS-AUTHENTIFIKATOR
504
INTEGRITÄTS-MESSUNGS-SELBST-TESTEINRICHTUNG

FIG. 18

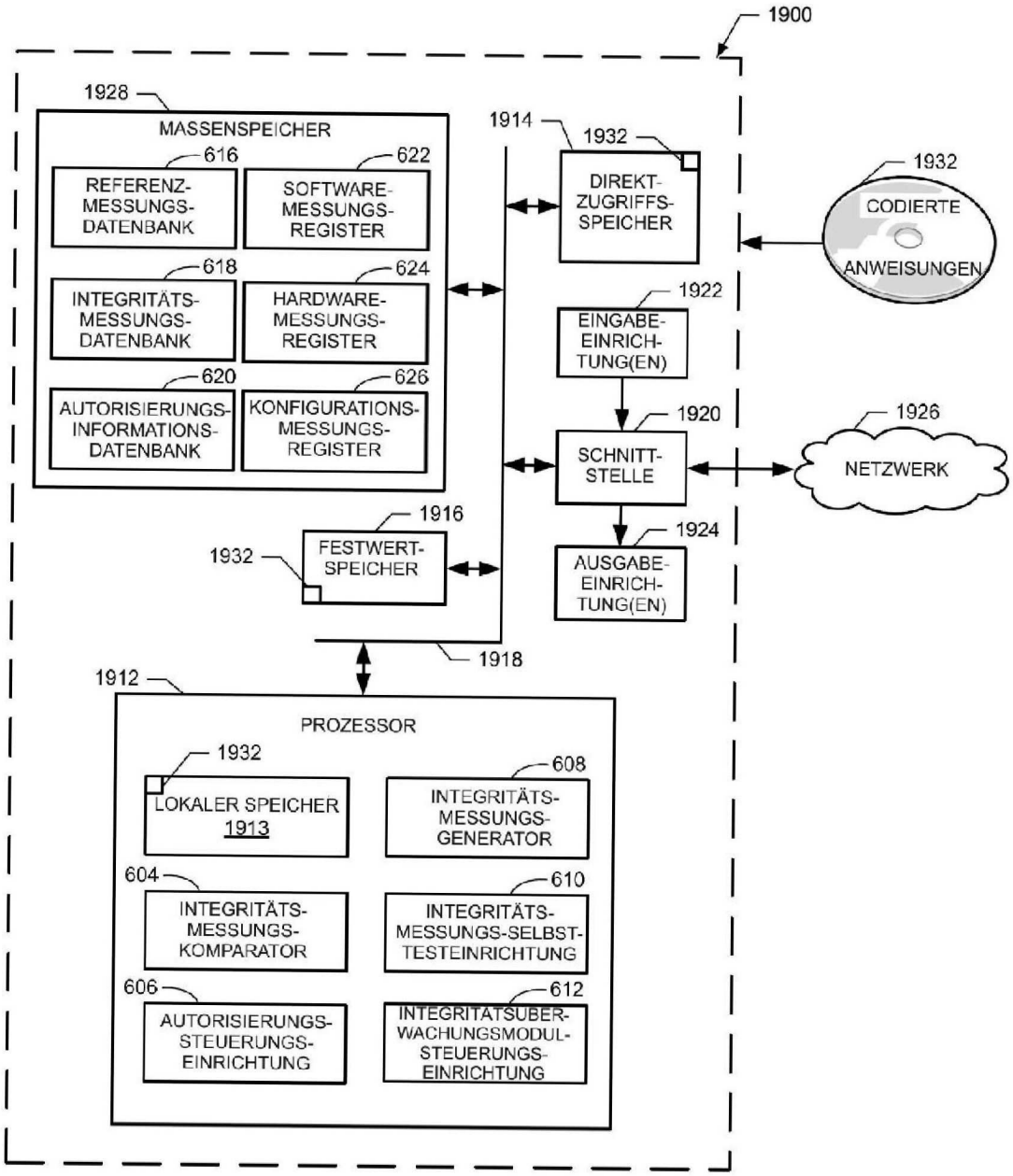


FIG. 19

 Das Dokument wurde durch die Firma Luminess hergestellt.