(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 103 25 816 B4** 2008.09.04

(12) **Patentschrift**

(21) Aktenzeichen: **103 25 816.7**
(22) Anmeldetag: **06.06.2003**
(43) Offenlegungstag: **13.01.2005**
(45) Veröffentlichungstag der Patenterteilung: **04.09.2008**

(51) Int Cl.[8]: ***H04L 9/30*** (2006.01)

(73) Patentinhaber:
Hewlett-Packard Development Co., L.P., Houston, Tex., US

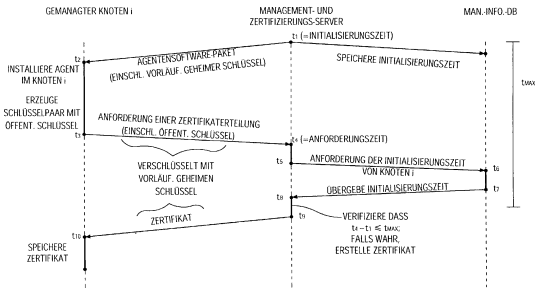
(74) Vertreter:
Samson & Partner, Patentanwälte, 80538 München

(72) Erfinder:
Bosler, Martin, 72827 Wannweil, DE

(56) Für die Beurteilung der Patentfähigkeit in Betracht gezogene Druckschriften:
US2003/00 70 078 A1
EP 10 96 446 A2
Schneier,B.: "Angewandte Kryptographie", Addison-Wesley, 1.Auflage 1996;
Klingst,H.: "Auf dem Weg zum Netz des Vertrauens - Grundlagen und Besonderheiten der Public Key-Verfahren". In: Networking, Datacom 5/2000, S.44-46;
hp Open View Quick Reference Guide, 2003;
Stevens, W.R.: "TCP/IP Illustrated", Bd. 1, 1994, S. 359-388;
Microsoft Windows 2000 Security Technical Referen-ce, Redmond 2000, S. 163 u. 175-208;
Tanenbaum, A.S.: "Computer Networks", 4. Aufl., 2003, Pearson Education International, S. 752-755, 760-762, 765-770, 798-799, 813-816;
Schneier, B.: "Applied Cryptography", John Wiley + Sons, 2. Aufl., 1996, S. 31-32, 37-39, 41-44;

(54) Bezeichnung: **Infrastruktur für öffentliche Schlüssel für Netzwerk-Management**

(57) Hauptanspruch: Verfahren zum Einrichten eines Managementagenten (4) in einem Knoten (2) eines IT-Netzwerks (1), in dem gemanagte Knoten (2) in der Management-Kommunikation authentisiert werden, wobei die Authentisierung auf Kryptographie mit öffentlichen Schlüsseln beruht, umfassend:
Einleitung einer automatisierten Installation eines Management-Agenten (4) auf einem Knoten (2) durch einen Management-Server (8) durch Übertragung der zu installierenden Agentensoftware und/oder einer Installationsanforderung vom Management-Server (8) auf den Knoten (2), wobei dies eine sog. Initialisierungszeit definiert;
Automatisierte Installation des Agenten (4) auf dem Knoten (2);
Anforderung der Erteilung eines Zertifikats (7) für einen öffentlichen Schlüssel durch den Agenten (4) bei einem Zertifizierungs-Server (13), wobei dies eine sog. Anforderungszeit definiert;
Automatische Erteilung des angeforderten Zertifikats (7) durch den Zertifzierungs-Server (13), falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit in einem maximalen Zeitintervall für automatische Zertifikaterteilung liegt, oder Verweigerung der automatischen Erteilung des angeforderten Zertifikats (7) durch den Zertifizierungs-Server (13), falls das Zeitintervall...



Beschreibung

GEBIET DER ERFINDUNG

[0001] Die vorliegende Erfindung bezieht sich allgemein auf eine Infrastruktur für öffentliche Schlüssel für das Netzwerk-Management, und beispielsweise auf ein Verfahren zum Erteilen eines Zertifikats für einen öffentlichen Schlüssel, ein Computerprogramm zum Ausführen dieses Verfahrens und ein IT-Netzwerk-Managementsystem.

HINTERGRUND DER ERFINDUNG

[0002] Aufgrund der zunehmenden Komplexität moderner informationstechnischer (IT) Netzwerke sind integrierte Netzwerkmanagementsysteme ein wichtiges Werkzeug zum Einrichten und Betreiben von IT-Netzwerken geworden. Beispielsweise bietet Hewlett-Packard ein solches Netzwerkmanagementsystem unter dem Namen "hp OpenView" an (siehe beispielsweise hp OpenView Quick Reference Guide, Februar 2003). Es ist eine Aufgabe des "Netzwerk-Managements", den Status, die Leistungsfähigkeit oder Verfügbarkeit von Netzwerkelementen zu überwachen und beispielsweise einem Netzwerk-Operator oder einem Service-Verantwortlichen Überwachungsergebnisse zu liefern. Im allgemeinen schließt Netzwerküberwachung eine Alarmfunktion ein, die den Operator oder Service-Verantwortlichen im Fall eines zu beachtenden Zwischenfalls, beispielsweise eines Ausfalls, eines drohenden Ausfalls oder einer Leistungs- oder Verfügbarkeitsabnahme eines Netzwerkelements warnt. Üblicherweise (aber nicht zwingend) sieht ein Netzwerkmanagementsystem nicht nur einen Informationsfluß von den gemanagten Netzwerkelementen zu einem Management-Server und z. B. einem Netzwerk-Operator vor, sondern erlaubt auch, gemanagte Netzwerkelemente, z. B. durch den Netzwerk-Operator oder automatisch durch einen Netzwerkmanagementprozeß zu manipulieren. "Manipulieren" kann beispielsweise folgendes umfassen: Konfigurieren von Netzwerkelementen oder Ändern ihrer existierenden Konfiguration, Starten und Anhalten von Prozessen in Netzwerkelementen, Zuteilen von Speicher an Netzwerkelemente, usw.

[0003] Üblicherweise sind Netzwerkmanagementsysteme verteilte Systeme, die einerseits verteilte Managementsoftware umfassen, die auf den gemanagten Netzwerkelementen (auch "gemanagte Knoten" genannt) läuft. Die auf dem gemanagten Knoten laufende Managementsoftware wird "Agent" genannt. Andererseits umfaßt ein übliches Netzwerkmanagementsystem einen Management-Server, der mit den Agenten kommuniziert und ihnen übergeordnet ist. Die Kommunikation erfolgt im allgemeinen in zwei Richtungen: Der Management-Server erhält Information von den Agenten über die Funktion des gemanagten Knoten, zu dem der jeweilige Agent gehört, oder er sendet eine Anforderung oder eine Instruktion an einen Agenten. Beim OpenView-Managementsystem, welches ein derartiges verteiltes Managementsystem ist, leiten die Agenten nicht einfach die von ihrem Knoten erhaltene Information an den Management-Server weiter, sondern sie haben eine gewisse "Intelligenz". Ein Agent ist beispielsweise dazu eingerichtet, gemäß benutzer-definierten Regeln die Information zu filtern, die er von seinem gemanagten Knoten über SNMP-Anforderungen erhält oder die er in einer Log-File des gemanagten Knotens findet; er leitet nur die diejenige Information an den Management-Server weiter, welche dieses Filter passiert (eine Beschreibung von SNMP (Simple Network Management Protocol) findet sich beispielsweise bei W. Richard Stevens: TCP/IP Illustrated, Band 1, The Protocols, 1994, S. 359–388). Managementsysteme dieser Art sind beispielsweise in EP 1 244 251 A1 und EP 1 257 087 A1 beschrieben. Die Kommunikation zwischen den Agenten und dem Management-Server beruht nicht auf SNMP, sondern verwendet ein leistungsfähigeres Netzwerkmanagement-Protokoll, das dem Netzwerkmanagementsystem proprietär ist.

[0004] Obwohl Netzwerkmanagement oft innerhalb einer nicht-öffentlichen Netzwerk-Domäne lokalisiert ist, kann Netzwerksicherheit auch ein Gegenstand in Netzwerkmanagementsystemen sein. Beispielsweise ist die Verwendung von symmetrischer Verschlüsselung mit geheimen Schlüsseln im Rahmen von SNMP vorgeschlagen worden, um Benutzerauthentisierung und eine Nachrichtenintegritätskontrolle bereitzustellen. (Siehe U. Blumenthal et al.: User-Based Security Model (USN) for version 3 of the Simple Network Management Protocol (SNMPv3), Network Working Group, RFC2574, April 1999, http://www.ietf.org/rfc/rfc2574.txt). WO 01/24444 A2 beschreibt die Anwendung dieses Vorschlags auf ein Breitbandzugangs-Netzwerk. US 2003/0033521 beschreibt die Verwendung von Verschlüsselung mit öffentlichem Schlüssel, um eine Autorisierung zum Anfordern einer "Switch-User"-Operation im Rahmen eines Netzwerkmanagementsystems zu verifizieren.

[0005] Allgemein stellen wenigstens einige der gegenwärtig auf dem Markt befindlichen Betriebssysteme eine Nutzerauthentisierungs-Funktion und eine Infrastruktur für öffentliche Schlüssel bereit (siehe beispielsweise Microsoft Windows 2000 Security Technical Reference, Redmond 2000, S. 163 und 175–208). Um die Kommunikation über das Internet zu sichern, ist ein SSL (Secure Sockets Layer) genanntes Sicherheitsprotokoll entwickelt worden, das nun breite Anwendung findet. SSL baut eine sichere Verbindung zwischen zwei "Sockets" auf (Sockets sind Transportdienst-Primitive), welche eine gegenseitige Authentisierung von Client und Server umfaßt (siehe beispielsweise Andrew S. Tanenbaum: Com-

puter Networks, 4. Auflage, 2003, Pearson Education International, S. 813–816).

[0006] EP 1 096 446 A2 beschreibt ein Verfahren zur Erteilung von Zertifikaten an Bankterminals. Die Zertifikaterteilung erfolgt nach automatischer Überprüfung von IP-Adresse und Name des anfordernden Terminals durch einen Zertifikatserver. In periodischen Abständen wird überprüft, ob alle zur Zertifizierung berechtigten Terminals tatsächlich zertifiziert wurden. Die Identität eines zertifikatanfordernden Terminals kann überdies vor einer Zertifikaterteilung auf herkömmliche Weise durch einen Administrator überprüft werden.

ZUSAMMENFASSUNG DER ERFINDUNG

[0007] Gemäß einem Aspekt wird ein Verfahren zum Einrichten eines gemanagten Agenten in einem Knoten eines IT-Netzwerks bereitgestellt, in dem gemanagte Knoten in der Management-Kommunikation authentisiert werden, wobei die Authentisierung auf Kryptographie mit öffentlichem Schlüssel beruht. Das Verfahren umfaßt die Merkmale gemäß Anspruch 1.

[0008] Gemäß einem weiteren Aspekt wird ein IT-Netzwerk-Managementsystem bereitgestellt. Dieses umfaßt einen Management-Server und einen Zertifizierungsserver, und ist in Anspruch 11 näher definiert. Gemäß einem weiteren Aspekt wird ein Computerprogramm bereitgestellt, welches Programmcode umfaßt, der, wenn er in einem gemanagten IT-Netzwerk ausgeführt wird, zum Durchführen eines Verfahrens zum Einrichten eines Managementagenten in einem Knoten dient, wobei in dem IT-Netzwerk gemanagte Knoten in der Managementkommunikation authentisiert werden, und wobei die Authentisierung auf Kryptographie mit öffentlichem Schlüssel beruht. Der Programmcode ist gemäß Anspruch 12 eingerichtet.

[0009] Weitere Merkmale sind in den offenbarten Verfahren und Erzeugnissen implizit enthalten oder werden für den Fachmann aufgrund der folgenden detaillierten Beschreibung von Ausführungsformen und der angefügten Zeichnungen ersichtlich.

BESCHREIBUNG DER ZEICHNUNGEN

[0010] Ausführungsformen der Erfindung werden nun beispielhaft und unter Bezugnahme auf die angefügten Zeichnungen beschrieben, wobei:

[0011] Fig. 1 ein beispielhaftes gemanagtes IT-Netzwerk veranschaulicht;

[0012] Fig. 2 ein Zeitdiagramm ist, welches die Agenteninitialisierung und die Zertifikaterteilung veranschaulicht;

[0013] Fig. 3 ein Flußdiagramm des in Fig. 2 gezeigten Verfahrens der Agenteninitialisierung und Zertifikaterteilung ist;

[0014] Fig. 4 eine beispielhafte vereinfachte Darstellung eines Zertifikats für einen öffentlichen Schlüssel ist;

[0015] Fig. 5 ein Flußdiagramm einer authentisierten Managementkommunikation, bei deren Beginn eine sichere Sitzung errichtet wird, ist;

[0016] Fig. 6 ein Flußdiagramm einer anderen Ausführungsform einer authentisierten Managementkommunikation, bei der digitale Signaturen verwendet werden, ist;

[0017] Fig. 7 ein Flußdiagramm eines Verfahrens zur Verifizierung eines öffentlichen Schlüssels auf der Grundlage eines Zertifikats für den öffentlichen Schlüssel ist;

[0018] Fig. 8 eine diagrammatische Darstellung eines beispielhaften Computersystems des IT-Netzwerks zeigt, in dem ein Satz von Instruktionen ausgeführt werden kann, welche das Computersystem veranlassen, die hier diskutierten Verfahren durchzuführen.

BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN

[0019] Fig. 1 veranschaulicht ein beispielhaftes gemanagtes IT-Netzwerk. Bevor die detaillierte Beschreibung der Fig. 1 fortgesetzt wird, werden jedoch einige Punkte der Ausführungsformen besprochen.

[0020] In der Vergangenheit war bei der herkömmlichen Kryptographie mit symmetrischem Schlüssel das Verteilen der Schlüssel (die im folgenden konsistent als geheime Schlüssel bezeichnet werden) immer das schwächste Glied in den meisten Verschlüsselungssystemen. Wie stark auch ein Verschlüsselungssystem war, falls es einem Eindringling gelang, den Schlüssel zu stehlen, war das System wertlos. Schlüssel mußten vor Diebstahl geschützt werden, aber sie mußten auch an alle Benutzer des Systems verteilt werden. Dies war ein inhärentes, eingebautes Problem (siehe beispielsweise Tanenbaum, S. 752).

[0021] Die Notwendigkeit zur Verteilung geheimer Schlüssel wurde durch Verschlüsselung mit öffentlichem Schlüssel überwunden, bei welcher zwei verschiedene Schlüssel verwendet werden – ein öffentlicher und ein privater (diese Bezeichnungsweise wird im folgenden konsistent verwendet, im Gegensatz zu den geheimen Schlüsseln, die für die herkömmliche Verschlüsselung mit symmetrischem Schlüssel verwendet wird). Es ist rechnerisch schwierig, den privaten Schlüssel von dem öffentli-

chen Schlüssel abzuleiten. Jedermann mit dem öffentlichen Schlüssel kann eine Nachricht verschlüsseln, aber sie nicht entschlüsseln. Nur die Person (oder der Computer) mit dem privaten Schlüssel kann die Nachricht entschlüsseln. Im Gegensatz zur Verschlüsselung mit symmetrischem Schlüssel, braucht der private Schlüssel nicht verteilt zu werden, sondern kann in einem geschützten Bereich des Computers des Entschlüsselnden geheimgehalten werden. Mathematisch beruht dieser Prozeß auf sog. "Falltür-Einweg-Funktionen". Dies sind Funktionen, die in einer Richtung leicht, in der anderen Richtung jedoch schwer zu berechnen sind. Wenn die "Falltür" jedoch bekannt ist, kann man die Funktion leicht in der anderen Richtung berechnen. Wenn Geheimhaltung erzielt werden soll, ist die Verschlüsselung die einfache Richtung. Die Instruktionen für die Verschlüsselung sind der öffentliche Schlüssel; jedermann kann eine Nachricht verschlüsseln. Die Entschlüsselung ist die schwierige Richtung, die typischerweise so schwierig gemacht ist, daß selbst mit schnellen Computern viele Jahre (typischerweise Tausende von Jahren) benötigt werden, um die Nachricht ohne die "Falltür" zu entschlüsseln. Die Falltür ist der private Schlüssel (siehe Bruce Schneier, Applied Cryptography, John Wiley & Sons, 2. Auflage, 1996, Seiten 31 bis 32, und Tanenbaum, S. 752–753). Wenn eine Authentisierung erzielt werden soll, können die Rollen des privaten und des öffentlichen Schlüssels vertauscht werden: Eine Nachricht kann dann mit dem privaten Schlüssel des Absenders verschlüsselt und vom Empfänger mit dem öffentlichen Schlüssel des Absenders entschlüsselt werden.

[0022] Ein verbreiteter Algorithmus mit öffentlichem Schlüssel ist RSA, welcher auf der Schwierigkeit des Zerlegens großer Zahlen in Faktoren beruht. Andere Algorithmen mit öffentlichem Schlüssel beruhen auf der Schwierigkeit, diskrete Logarithmen modulo einer großem Primzahl zu berechnen. Weitere Algorithmen beruhen auf elliptischen Kurven (siehe Tanenbaum, S. 753–755).

[0023] Es sei nun angenommen, daß der Zweck der Verwendung von Verschlüsselung mit öffentlichem Schlüssel Geheimhaltung ist, d. h. daß beispielsweise Alice eine geheime Nachricht an Bob senden möchte. Es folgt ein Verschlüsselungsprotokoll, mit dem geheime Übertragung erzielt werden kann (zum Zweck der Veranschaulichung wird auch hier die verbreitete "Alice-Bob"-Sprache verwendet, obwohl in den Ausführungsformen Nachrichten hauptsächlich zwischen Maschinen (z. B. Computern) statt Personen ausgetauscht werden; Alice und Bob können somit als Namen von Maschinen betrachtet werden):

(1) Bob sendet Alice seinen öffentlichen Schlüssel, oder Alice erhält Bobs öffentlichen Schlüssel von einer (öffentlich zugreifbaren) Datenbasis.

(2) Alice verschlüsselt ihre Nachricht unter Verwendung von Bobs öffentlichem Schlüssel und sendet sie an Bob.

(3) Bob entschlüsselt Alice' Nachricht unter Verwendung seines privaten Schlüssels.

[0024] Kein geheimer Schlüssel muß verteilt werden, niemand anderer kann die verschlüsselte Nachricht lesen, da es zu schwierig ist, Bobs privaten Schlüssel von seinem öffentlichen bekannten öffentlichen Schlüssel abzuleiten. Somit wird ein sicherer Kanal zwischen Alice und Bob errichtet (siehe Schneier, S. 31–32).

[0025] Neben Geheimhaltung kann Verschlüsselung mit öffentlichem Schlüssel dazu verwendet werden, die Authentizität und Integrität eines Dokuments durch Verwendung einer sog. "digitalen Signatur" sicherzustellen. Das grundlegende Authentisierungsprotokoll ist einfach:

(1) Alice verschlüsselt das Dokument mit ihrem privaten Schlüssel, wodurch sie es signiert.

(2) Alice sendet das signierte Dokument (d. h. die digitale Signatur) an Bob.

(3) Bob entschlüsselt das Dokument mit Alice' öffentlichem Schlüssel, wodurch er die Signatur verifiziert.

[0026] Nur Alice kennt ihren privaten Schlüssel; wenn Bob das Dokument mit Alice' öffentlichem Schlüssel verifiziert, dann weiß er, daß sie es signiert hat ("Authentizität"). Das signierte Dokument ist unveränderlich; falls es irgendeine Änderung an dem Dokument gibt, kann die Signatur nicht mehr mit Alice' öffentlichem Schlüssel verifiziert werden ("Integrität") (siehe Schneier, S. 37–38).

[0027] Algorithmen mit öffentlichem Schlüssel sind oft relativ ineffizient, insbesondere, wenn lange Dokumente zu signieren sind. Um Zeit zu sparen sind bei manchen Ausführungsformen der oben beschriebenen Infrastruktur mit öffentlichem Schlüssel Protokolle für digitale Signatur mit Einweg-Hash-Funktionen implementiert. Eine Einweg-Hash-Funktion ist eine Kompressionsfunktion (die oft "Message Digest" genannt wird). Eine Hash-Funktion nimmt einen Eingabe-String variabler Länge (genannt "Vor-Bild") auf, und wandelt ihn in einen (im allgemeinen kürzeren) Ausgabe-String um, der im allgemeinen eine feste Länge hat (genannt "Hash-Wert"). Eine Einweg-Hash-Funktion ist eine Hash-Funktion, die in einer Richtung arbeitet: Es ist leicht, einen Hash-Wert aus einem Vor-Bild zu berechnen, aber es ist schwierig, ein Vor-Bild zu erzeugen, das einen bestimmten Hash-Wert ergibt. Eine gute Einweg-Hash-Funktion ist auch kollisionsfrei, was bedeutet, daß es schwierig ist, zwei Vor-Bilder mit gleichem Hash-Wert zu erzeugen (siehe Schneier, S. 30–31). Die verbreitetsten Einweg-Hash-Funktionen sind MD5 und SHA-1 (siehe Tanenbaum, S. 760–762).

[0028] Bei einigen der Ausführungsformen beruht

die Authentisierung auf Hash-Werten von Dokumenten, aber nicht auf den Dokumenten selbst. Das oben beschriebene grundlegende Authentisierungsprotokoll ist dann modifiziert: Statt ein Dokument zu signieren, wird ein Hash-Wert eines Dokuments signiert (siehe Schneier, S. 38–39 und 41–44):

(1) Alice erzeugt einen Einweg-Hash-Wert eines Dokuments.

(2) Alice verschlüsselt den Hash-Wert mit ihrem privaten Schlüssel, wodurch sie das Dokument signiert.

(3) Alice sendet das Dokument und den signierten Hash-Wert (d. h. den verschlüsselten Hash-Wert) an Bob.

(4) Bob erzeugt einen Einweg-Hash-Wert des von Alice gesendeten Dokuments. Er entschlüsselt dann den signierten Hash-Wert mit Alice' öffentlichem Schlüssel unter Verwendung des Algorithmus für digitale Signatur. Wenn der signierte Hash-Wert mit dem von ihm erzeugten Hash-Wert übereinstimmt, ist die Signatur gültig.

[0029] Es gibt jedoch ein Problem bei der Verschlüsselung mit öffentlichem Schlüssel: Auf welche Weise bekommen Alice und Bob den öffentlichen Schlüssel des jeweils anderen, um den Kommunikationsprozeß zu beginnen? Eine Lösung wäre die Veröffentlichung des Schlüssels irgendwo in einem Netzwerk, so daß er von allen interessierten Teilnehmern durch eine Art GET-Anforderung erhalten werden kann. Ein Eindringling (genannt "Trudy") könnte jedoch die Anforderung abfangen und mit einem falschen öffentlichen Schlüssel (z. B. Trudy's öffentlichem Schlüssel) antworten. Trudy könnte dann einen Einweg-Hash-Wert eines Dokuments erzeugen, ihn mit ihrem privaten Schlüssel verschlüsseln und das Dokument und den signierten Hash-Wert an Bob senden. Bob würde den signierten Hash-Wert mit dem Alice zugeschriebenen öffentlichen Schlüssel entschlüsseln (welcher tatsächlich Trudys öffentlicher Schlüssel ist) und würde glauben, daß Alice das Dokument signiert hat. Trudy könnte auch ein von Alice an Bob gesendetes Dokument mit einer Signatur abfangen, es modifizieren, einen Einweg-Hash-Wert des modifizierten Dokuments erzeugen, den Hash-Wert mit Trudys privatem Schlüssel verschlüsseln und das Dokument und den signierten Hash-Wert an Bob senden. Bob würde den signierten Hash-Wert mit dem Alice zugeschriebenen öffentlichen Schlüssel entschlüsseln und würde glauben, daß das Dokument ungeändert ist (siehe Tanenbaum, S. 765).

[0030] Eine Möglichkeit zur sicheren Verteilung öffentlicher Schlüssel beruht auf Zertifikaten für öffentliche Schlüssel. Eine Organisation oder ein Computer, welche(r) öffentliche Schlüssel zertifiziert, wird "CA" (Certification Authority) genannt. Im allgemeinen zertifiziert ein Zertifikat, daß ein bestimmter öffentlicher Schlüssel zu einer bestimmten Person oder einem bestimmten Netzwerkknoten gehört. Die CA hat selbst ein Schlüsselpaar mir privatem und öffentlichem Schlüssel, und das Zertifikat enthält einen Hash-Wert des Zertifikats, der mit dem privaten Schlüssel der CA signiert ist. Es sei angenommen, daß der öffentliche Schlüssel der CA allen an der Kommunikation teilnehmenden Personen oder Knoten bekannt ist, und daß er daher nicht von einem Eindringling durch einen falschen Schlüssel ersetzt werden kann. Ein Teilnehmer, der ein solches Zertifikat für einen öffentlichen Schlüssel erhält, verifiziert die Authentizität und Integrität des Zertifkats für den öffentlichen Schlüssel, und dadurch des öffentlichen Schlüssels selbst, durch das oben beschriebene Verifizierungsprotokoll, d. h. durch Erzeugen eines Einweg-Hash-Werts des Zertifikats und Entschlüsseln des signierten Hash-Werts mit dem öffentlichen Schlüssel der CA. Wenn der signierte Hash-Wert mit dem erzeugten Hash-Wert übereinstimmt, ist die Signatur gültig, d. h. ist der zertifizierte öffentliche Schlüssel authentisch und ungeändert (siehe Tanenbaum, S. 765–767). In größeren Netzwerken kann statt nur einer CA eine Hierarchie von CAs gegeben sein, in der eine CA erster Ordnung CAs zweiter Ordnung zertifiziert, welche wiederum weitere CA-Ordnungen zertifizieren, wobei die letzte Ordnung wirkliche CAs darstellen kann, welche Zertifikate ausgeben (siehe Tanenbaum, S. 768–770). Um freien Austausch und freie Verwendung der Zertifikate für öffentliche Schlüssel zwischen Teilnehmern zu ermöglichen, ist das Zertifkatformat genormt. Die Norm wird X.509 genannt und findet im Internet weite Verbreitung (siehe Tanenbaum, S. 767–768).

[0031] Statt eine digitale Signatur (d. h. ein signiertes Dokument oder einen signierten Hash-Wert) auszutauschen, erfolgt bei einigen der Ausführungsformen die Kommunikation zwischen zwei kommunizierenden Teilnehmern innerhalb einer sicheren Sitzung (Session), die durch Authentisierung wenigstens eines der Teilnehmer oder durch gegenseitige Authentisierung beider Teilnehmer auf der Grundlage von Kryptographie mit öffentlichem Schlüssel eingerichtet wird. Ein Beispiel einer Einrichtung einer sicheren Sitzung mit gegenseitiger Authentisierung ist (siehe Tanenbaum, S. 798–799):

(1) Alice empfängt Bobs öffentlichen Schlüssel, z. B. von Bob oder einem Verzeichnis für öffentliche Schlüssel;

(2) Alice sendet Bob eine Nachricht, welche eine Zufallszahl (genannt "Nonce") enthält; Alice verschlüsselt die Nachricht mit Bobs öffentlichem Schlüssel (aber nicht mit Alice' privatem Schlüssel, wie bei den obigen Digital-Signatur-Protokollen).

(3) Bob erhält Alice' öffentlichen Schlüssel, z. B. von Alice oder einem Verzeichnis für öffentliche Schlüssel.

(4) Bob entschlüsselt die Nachricht, welche die Nonce von Alice enthält, und sendet eine Nach-

richt an Alice, welche die Nonce von Alice sowie eine von ihm gewählte Nonce und einen von ihm gewählten vorgeschlagenen Sitzungs-Schlüssel enthält; Bob verschlüsselt diese Nachricht mit Alice' öffentlichem Schlüssel.
(5) Alice entschlüsselt Bobs Nachricht unter Verwendung ihres privaten Schlüssels; wenn sie ihre Nonce darin erkennt, weiß sie, daß Bob ihre Nachricht erhalten hat, da niemand sonst die Möglichkeit hat, ihre Nonce herauszufinden.
(6) Alice verschlüsselt Bobs Nonce mit dem Sitzungs-Schlüssel (nun unter Verwendung symmetrischer Verschlüsselungstechniken) und sendet dies an Bob.
(7) Bob entschlüsselt diese Nachricht mit dem Sitzungs-Schlüssel; wenn er seine Nonce erkennt, weiß er, daß Alice seine Nachricht erhalten hat.

[0032] Damit ist nun ein geheimer (symmetrischer) Schlüssel definiert, den nur Alice und Bob kennen. Nun kann eine sichere Sitzung durchgeführt werden, indem die während der Sitzung ausgetauschten Nachrichten mit dem geheimen Sitzungsschlüssel verschlüsselt werden (unter Verwendung symmetrischer Ent- und Verschlüsselung). Da nur Alice und Bob den Sitzungsschlüssel kennen, verifiziert die Tatsache, daß eine Nachricht durch den Sitzungsschlüssel entschlüsselt wurde, die Authentizität des Absenders, der entweder Alice oder Bob ist. Wiederum könnte ein Eindringling (Trudy) die Kommunikation zwischen Alice und Bob abfangen und Alice glauben machen, daß sie mit Bob statt mit Trudy kommuniziert. Dies ist wiederum dadurch ausgeschlossen, daß verlangt wird, daß Bobs öffentlicher Schlüssel und auch Alice' öffentlicher Schlüssel durch Zertifikate für die öffentlichen Schlüssel verifiziert werden.

[0033] Um derartige sichere Verbindungen zwischen Teilnehmern im Internet zu ermöglichen, ist ein SSL (Secure Sockets Layer) genanntes Protokoll eingeführt worden. Das SSL-Protokoll ist nun weit verbreitet, und SSL-Software ist als Open-Source-Software erhältlich (z. B. OpenSSL). Wenn HTTP über SSL verwendet wird, wird es HTTPS (sicheres HTTP) genannt. Eine jüngere SSL-Version (Version 3.1) wird auch TLS (Transport Layer Security) genannt; sie ist in RFC2246 beschrieben (T. Dierks et al., The TLS Protocol Version 1.0, January 1999, http://www.ietf.org/rfc/rfc2246.txt). In einigen Ausführungsformen wird das SSL-Protokoll verwendet, um eine sichere Verbindung ("Sitzung") zwischen einem ersten Managementelement (einem gemanagten Knoten) und einem zweiten Managementelement (z. B. einem Management-Server oder einem anderen gemanagten Knoten) zu errichten. Bei einer solchen sicheren Verbindung wird zuerst eine Parameterabstimmung zwischen dem ersten Managementelement und dem zweiten Managementelement durchgeführt. Dann erfolgt eine Authentisierung des ersten Managementelements und, vorzugsweise, des zweiten Managementelements, d. h. eine gegenseitige Authentisierung. Die Authentisierung ist für die gesamte Sitzung gültig. Dies wird erreicht, indem alle während der Sitzung ausgetauschten Nachrichten auf der Grundlage symmetrischer Verschlüsselung verschlüsselt werden. Der verwendete geheime Schlüssel wird während der Authentisierungsphase zwischen dem ersten und zweiten Managementelement ausgetauscht, und zwar verschlüsselt mit dem öffentlichen Schlüssel des ersten und/oder zweiten Managementelements. Die Authentizität der öffentlichen Schlüssel wird durch Zertifikate für die öffentlichen Schlüssel verifiziert, wie oben erläutert wurde. Die Datenintegrität ist ebenfalls während der gesamten Sitzung geschützt. Für Einzelheiten über SSL, siehe beispielsweise Tanenbaum, S. 813–816. Natürlich kann statt SSL jedes andere (proprietäre oder genormte) Protokoll verwendet werden, das gesicherte Verbindungen bereitstellt.

[0034] In den Ausführungsformen schließen die während einer solchen Sitzung ausgetauschten Nachrichten Managementnachrichten ein, welche eine Managementkommunikation zwischen gemanagten Elementen eines IT-Netzwerkes bilden. Die Managementelemente umfassen einen Managementserver und gemanagte Knoten. Bei manchen der Ausführungsformen schließt das Management des Netzwerks ein Überwachen des Status, der Leistungsfähigkeit oder Verfügbarkeit der gemanagten Knoten und das Liefern von Überwachungsergebnissen, beispielsweise an einen Netzwerk-Operator oder einen Service-Verantworlichen ein. Bei einigen der Ausführungsformen schließt die Überwachung einer Alarmfunktion ein, welche den Operator oder den Service-Verantwortlichen im Fall eines Zwischenfalls, der Aufmerksamkeit erfordert, warnt, beispielsweise eines Ausfalls, eines drohenden Ausfalls oder einer Abnahme der Leistungsfähigkeit oder Verfügbarkeit eines gemanagten Knotens. Das Netzwerkmanagementsystem erbringt nicht nur einen Informationsfluß von den gemanagten Knoten zum Managementserver und, z. B. zum Netzwerk-Operator, sondern ermöglicht auch ein Manipulieren der gemanagten Knoten, z. B. durch den Netzwerk-Operator oder automatisch durch einen Netzwerkmanagementprozeß. Bei einigen der Ausführungsformen umfaßt "Manipulieren" Konfigurieren eines gemanagten Knotens oder Ändern seiner existierenden Konfiguration, Starten und Anhalten von Prozessen in einem gemanagten Knoten, Zuteilen von Speicher an einen gemanagten Knoten, usw. Das Netzwerkmanagementsystem ist ein verteiltes System, welches einerseits verteilte Managementsoftware umfaßt, die auf den gemanagten Knoten läuft; diese Software wird "Agent" genannt. Andererseits kommuniziert der Managementserver mit den Agenten und ist ihnen übergeordnet. Bei einigen der Ausführungsformen erfolgt die Kommunikation in zwei Richtungen: Der Managementserver empfängt Nachrichten von den Agenten,

welche Information über die Funktion des gemanagten Knotens, dem der jeweilige Agent zugeordnet ist, enthält, oder er sendet Nachrichten, die eine Anforderung oder eine Instruktion enthalten, an den Agenten. Bei einigen der Ausführungsformen können Managementnachrichten auch direkt von einem ersten gemanagten Knoten an einen zweiten gesendet werden; wenn beispielsweise ein Prozeß auf dem zweiten gemanagten Knoten vom Status des ersten gemanagten Knotens abhängt, kann der erste gemanagte Knoten direkt Managementnachrichten mit Statusinformation des ersten gemanagten Knotens an den zweiten gemanagten Knoten senden. Bei einigen der Ausführungsformen leiten die Agenten nicht einfach Information von ihrem gemanagten Knoten an den Managementserver oder einen anderen gemanagten Knoten weiter, sondern sie sind vielmehr dazu eingerichtet, die Information zu filtern, die sie von ihrem gemanagten Knoten z. B. mittels SNMP-Anforderungen erhalten oder in einer Log-File des entsprechenden gemanagten Knotens finden. Das Filtern wird vorzugsweise entsprechend nutzer-konfigurierbarer Regeln durchgeführt. Die Agenten leiten nur diejenige Information an den Managementserver oder andere gemanagte Knoten weiter, welche dieses Filter passiert. Bei den Ausführungsformen beruht die Managementkommunikation zwischen den Agenten und dem Managementserver nicht auf SNMP, sondern verwendet vorzugsweise ein leistungsfähigeres Netzwerkmanagementprotokoll, das beispielsweise ein dem Netzwerkmanagementsystem proprietäres Protokoll ist.

[0035] Obwohl bei einigen der Ausführungsformen das Netzwerkmanagement innerhalb einer nicht-öffentlichen Domäne von Netzwerken lokalisiert ist, wird selbst bei diesen Ausführungsformen in der Managementkommunikation eine Managementelement-Authentisierung durchgeführt, die auf Verschlüsselung mit öffentlichem Schlüssel beruht. Zu diesem Zweck ist jedem der Managementelemente ein Schlüsselpaar mit privatem und öffentlichem Schlüssel zugeordnet. Es sei angenommen, daß eine Managementnachricht von einem ersten Managementelement an ein zweites Managementelement zu senden ist. Dann wird in einigen der Ausführungsformen eine sichere Sitzung eingerichtet, welche eine gegenseitige Authentisierung beider Managementelemente unter Verwendung ihrer öffentlichen Schlüssel einschließt. Die Managementnachricht wird dann während der sicheren Sitzung versendet. Bei anderen Ausführungsformen wird die Nachricht selbst durch eine digitale Signatur authentisiert. Die digitale Signatur wird von dem ersten Managementelement unter Verwendung des privaten Schlüssels des ersten Managementelements (d. h. dessen eigenen privaten Schlüssels) erzeugt. Das erste Managementelement sendet dann die Managementnachricht an das zweite Managementelement. Das zweite Managementelement verifiziert wiederum die Authentizität des Absenders durch Verwendung des öffentlichen Schlüssels des Absenders (hier: des öffentlichen Schlüssels des ersten Managementelements). Bei beiden Arten von Ausführungsformen umfaßt das Verifizieren der Authentizität des Managementelements ein Verifizieren der Authentizität von dessen öffentlichem Schlüssel durch ein Zertifikat dieses öffentlichen Schlüssels. Der öffentliche Schlüssel und/oder das Zertifikat für den öffentlichen Schlüssel können beispielsweise an das andere Managementelement gesendet werden, entweder zusammen mit einer Nonce, einem Sitzungsschlüssel, oder der zu authentisierenden Managementnachricht, oder separat. Alternativ können der öffentliche Schlüssel und/oder das Zertifikat für den öffentlichen Schlüssel irgendwo im Netzwerk gespeichert (z. B. in einem Verzeichnis für öffentliche Schlüssel und Zertifikate) und von dem jeweiligen Managementelement zur Verschlüsselung/Entschlüsselung von Nachrichten und zum Verifizieren von öffentlichen Schlüsseln geholt werden.

[0036] Bevor dem ersten Managementelement ein Zertifikat erteilt wird, ist eine Authentisierung auf der Grundlage von Verschlüsselung mit öffentlichem Schlüssel noch nicht für dieses verfügbar. Aber bereits in dieser Vor-Phase findet bereits Kommunikation mit dem Management- und/oder Zertifizierungsserver statt, beispielsweise Kommunikation, welche eine Installierung des Agenten des Managementelements veranlaßt. Um auch diese Vor-Kommunikation mit einer gewissen (reduzierten) Sicherheit auszustatten, werden bei manchen Ausführungsformen die in dieser Vor-Kommunikations-Phase ausgetauschten Daten mit einem, an das Managementelement gesendeten symmetrischen Schlüssel verschlüsselt. Der Schlüssel wird beispielsweise in "hartcodierter" Form in Binärdaten von komprimierten Softwarepaketen versendet, die zu Beginn der Vor-Kommunikations-Phase ausgetauscht werden, so daß die Schlüssel auf beiden Seiten nach einer Übersendung und Installierung der komprimierten Softwarepakete verfügbar sind.

[0037] In den Ausführungsformen wird die Aufgabe der CA von einem Zertifizierungsserver übernommen. "Zertifizierungsserver" ist als funktioneller Ausdruck zu verstehen: Es kann sich bei ihm um eine separate Maschine handeln, die der Erteilung von Zertifikaten mit öffentlichem Schlüssel gewidmet ist, oder um einen Prozeß mit dieser Funktion in einer Maschine, die auch anderen Zwecken dient, beispielsweise dem Managementserver. Der Zertifizierungsserver selbst hat ein Schlüsselpaar mit privatem und öffentlichem Schlüssel. Die von dem Zertifizierungsserver erteilten Zertifikate für öffentliche Schlüssel enthalten eine digitale Signatur des Zertifizierungsservers, die mit dem privaten Schlüssel des Zertifizierungsserver erzeugt wird. Das Verifizieren der Authentizität des öffentlichen Schlüssels eines Netzwerkelements um-

faßt ein Verifizieren des Zertifizierungsservers digitaler Signatur des Zertifikats unter Verwendung des öffentlichen Schlüssels des Zertifizierungsservers.

[0038] Die Aufgabe eines Zertifikats für einen öffentlichen Schlüssel liegt darin, den öffentlichen Schlüssel an einen bestimmten Inhaber (z. B. eine Person, ein Unternehmen, oder, hier, ein Managementelement) zu binden. Normalerweise würde die Erteilung eines Zertifikats auf eine, von einem gemanagten Knoten empfangene Anforderung eine menschliche Interaktion erfordern; beispielsweise würde ein Administator die Zertifikatanforderung gewähren. Dies liegt daran, daß die Herausgabe eines Zertifikats für einen öffentlichen Schlüssel eine sicherheitssensitive Aufgabe ist. Ein automatisches Erteilen von Zertifikaten auf Anforderung könnte daher ein inakzeptables "Sicherheitsloch" erzeugen.

[0039] In den Ausführungsformen ist eine andere Vorgehensweise verwirklicht: Zertifikate für öffentliche Schlüssel werden automatisch auf Anforderung von einem gemanagten Knoten hin herausgegeben, vorausgesetzt, daß das Zeitintervall zwischen einer Initialisierungszeit des gemanagten Knotens und einer der Anforderung zugeordneten Anforderungszeit (genannt "Initialisierungs-Anforderungs-Zeitintervall") in einem vordefinierten maximalen Zeitintervall liegt. Die "Initialisierungszeit" ist beispielsweise derjenige Zeitpunkt, an dem der Agent auf dem gemanagten Knoten installiert wurde. Dies ermöglicht, das Ausmaß von nötiger menschlicher Interaktion auf solche Fälle zu reduzieren, bei denen das Initialisierungs-Anforderungs-Zeitintervall länger als das maximale Zeitintervall ist. Andererseits gibt es ein "Sicherheitsloch" nur während der (relativ kurzen) Initialisierungsphase, was für typische Netzwerkmanagementanwendungen akzeptabel ist.

[0040] In den Ausführungsformen werden Managementagenten automatisch durch den Managementserver eingerichtet: Der Managementserver übersendet ein Managementagent-Softwarepaket an den betreffenden Knoten. Nach Empfang des Pakets wird der Agent in einer Selbstinstallierungsprozedur auf dem Knoten installiert. Bei einigen der Ausführungsformen ist die "Initialisierungszeit" derjenige Zeitpunkt, zu dem der Managementserver das Agent-Softwarepaket an den Agenten übersendet, oder er an den Agenten eine Anforderung sendet, die Selbstinstallierungsprozedur nun tatsächlich zu starten. In anderen Ausführungsformen sendet der Agent eine Bestätigung an den Managementserver, die anzeigt, daß er die Selbstintallierungsprozedur gestartet oder vollendet hat; bei einigen der Ausführungsformen ist die "Initialisierungszeit" derjenige Zeitpunkt, zu dem der Managementserver eine solche Bestätigung erhält. Während der Installierungsprozedur wird auch ein Schlüsselpaar mit privatem und öffentlichem Schlüssel bei dem Knoten erzeugt. Der private Schlüssel wird sicher in dem Knoten gespeichert. Am Ende der Installierungsprozedur bemerkt der Agent, daß sein öffentlicher Schlüssel noch nicht zertifiziert ist. Folglich sendet er eine Zertifikaterteilungsanforderung an den Zertifizierungsserver (der ein kombinierter Management/Zertifizierungsserver sein kann, wie oben erwähnt wurde). In den Ausführungsformen wird der Zeitpunkt des Empfangs dieser Anforderung als die "Anforderungszeit" angesehen. Wenn das auf diese Weise bestimmte Initialisierungs-Anforderungs-Zeitintervall in dem maximalen Zeitintervall liegt, wird die Anforderung gewährt und das angeforderte Zertifikat für den öffentlichen Schlüssel herausgegeben. Es wird an den Knoten gesendet, von dem die Anforderung kam, und kann außerdem in einem Verzeichnis gespeichert werden (in welchem auch der zu der Anforderung gehörende öffentliche Schlüssel gespeichert werden kann).

[0041] In den Ausführungsformen wird die Initialisierungszeit als eine Eigenschaft des gemanagten Knotens gespeichert, z. B. in einer Managementinformations-Datenbasis. Wenn der Zertifizierungsserver später eine Zertifikaterteilungsanforderung von diesem Knoten erhält, ruft er die gespeicherte Initialisierungszeit ab und berechnet aus ihr das Initialisierungs-Anforderungs-Zeitintervall. Wenn Maschinen mit unterschiedlichen Uhren zum Bestimmen der Initialisierungszeit und der Anforderungszeit verwendet werden, könnte ein Fehler eingeführt werden, falls die beiden Uhren nicht synchron gehen. Daher sind in solchen Ausführungsformen die verschiedenen Uhren synchronisiert, beispielsweise durch Verwendung einer Uhrensynchronisierungsprozedur auf der Grundlage des Simple Network Time Protocol (siehe beispielsweise D. Mills: Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI, October 1996, http://www.ee.cis.udel.edu/~mills/database/rfc/rfc2030.text). In anderen Ausführungsformen wird ein und dieselbe Computeruhr verwendet, um die Initialisierungszeit und die Anforderungszeit zu bestimmen, z. B. in Ausführungsformen, in denen der Zertifizierungsserver und der Managementserver auf derselben Maschine laufende Prozesse sind. Bei solchen Ausführungsformen wird das Initialisierungs-Anforderungs-Zeitintervall auch dann korrekt bestimmt, wenn die Uhr falsch geht, da das Zeitintervall eine Differenz zwischen zwei absoluten Zeitmessungen ist.

[0042] Das maximale Zeitintervall für automatische Zertifikaterteilung ist konfigurierbar, beispielsweise durch einen Verwalter oder Operator des IT-Netzwerks. Das gewählte maximale Zeitintervall wird etwas länger als die typische Zeit sein, die für die Übertragung des Agent-Softwarepakets an den Knoten und die Selbstinstallierung des Agenten auf dem Knoten benötigt wird; ein typisches maximales Zeitintervall kann in der Größenordnung von fünf Minuten betragen.

[0043] Die Ausführungsformen des Computerprogramms mit Programmcode zum Ausführen der beschriebenen Verfahren umfassen jegliches maschinenlesbare Medium, das zum Speichern oder Codieren des Programmcodes geeignet ist. Der Begriff "maschinenlesbares Medium" soll folglich Festkörperspeicher, optische und magnetische Speichermedien und Trägerwellensignale umfassen, aber nicht auf solche beschränkt sein. Der Programmcode kann Maschinencode oder ein anderer Code sein, der durch Kompilierung und/oder Interpretierung in Maschinencode umgewandelt werden kann, wie Quellcode in einer höheren Programmiersprache, wie z. B. C++, oder in jeder anderen geeigneten imperativen oder funktionalen Programmiersprache, oder virtueller Maschinencode.

[0044] Die Hardware, auf der der Programmcode ausgeführt wird, sind Computersysteme, welche den Zertifizierungsserver, den Managementserver (wobei diese kombinierte sein können) und die gemanagten Knoten darstellen. Die Computersysteme umfassen beispielsweise einen Prozessor und einen Hauptspeicher, die miteinander über einen Bus kommunizieren, sowie Netzwerkschnittstelleneinrichtungen und, fakultativ, weitere Speicher- und Eingabe-/Ausgabe-Einrichtungen. Der Programmcode kann im Prozessor, im Hauptspeicher und/der den weiteren Speichereinrichtungen gespeichert sein und kann über die Netzwerkschnittstelleneinrichtung gesendet oder empfangen werden.

[0045] Nun zurückkehrend zu Fig. 1 zeigt diese ein beispielhaftes gemanagtes IT-Netzwerk **1**. Das Netzwerk **1** umfaßt drei gemanagte Knoten **2a**, **2b**, **2c**. Die gemanagten Knoten können beispielsweise Netzwerkverbindungsgeräte, wie Bridges, Switches, Router und/oder Endgeräte, wie Arbeitsplatzcomputer, Workstations, Server, usw. sein. In einem allgemeineren Sinn kann ein "gemanagter Knoten" auch eine gemanagte Applikation sein, wobei mehr als eine solche gemanagte Applikation auf einer Maschine laufen kann. Jeder gemanagte Knoten **2** ist durch einen Identifikator ID identifiziert, der in einem Identifikatorfeld **3** des gemanagten Knotens **2** gespeichert ist. Jeder gemanagte Knoten **2** hat einen Agenten **4**, der für lokale managementbezogene Verarbeitung und Kommunikation mit anderen Managementelementen des IT-Netzwerks **1** verantwortlich ist. In dem Beispiel der Fig. 1 ist der dritte gemanagte Knoten **2c** in einer Initialisierungsphase gezeigt, in welcher dessen Agent noch nicht installiert ist. Ein privater Schlüssel **5**, ein öffentlicher Schlüssel **6** und ein Zertifikat **7** für den öffentlichen Schlüssel sind in dem gemanagten Knoten **2** gespeichert, wobei der private Schlüssel **5** in einer sicheren Weise gespeichert ist, in der er nur für einen Benutzer sichtbar ist, für den der Agent **4** läuft, beispielsweise für einen Netzwerkadministrator. Der dritte gemanagte Knoten **2c** hat noch keine Schlüssel, und der zweite Knoten **2b** ist in einem Zwischenzustand gezeigt, in dem er bereits einen privaten und einen öffentlichen Schlüssel **5**, **6** hat, aber noch kein Zertifikat für den öffentlichen Schlüssel.

[0046] Ein Managementserver **8** ist das zentrale Gegenstück zu den verteilten Agenten **4**. Er empfängt Information von den Agenten **4** über die Funktion der jeweiligen gemanagten Knoten **2**, und sendet Managementanforderungen oder -instruktionen an die Agenten **4**. Der Managementserver **8** hat auch einen privaten Schlüssel, einen öffentlichen Schlüssel und ein Zertifikat für den öffentlichen Schlüssel, die in Fig. 1 mit "**9**", "**10**" und "**11**" bezeichnet sind.

[0047] Eine Managementinformations-Datenbasis **12** speichert managementbezogene Daten. Wie in Fig. 1 dargestellt ist, ist unter diesen Daten eine Tabelle mit einem Datensatz für jeden gemanagten Knoten **2** gespeichert, der durch den ID des gemanagten Knotens identifiziert ist. Eines der Datensatzattribute dieser Tabelle ist die Initialisierungszeit des betreffenden Agenten **4**. Auf die Managementinformations-Datenbasis **12** kann vom Managementserver **8** sowie den Agenten **4** und anderen Managementelementen, wie dem Zertifizierungsserver **13**, zugegriffen werden. Jedoch hat nur der Managementserver **8** Schreibzugriff auf die oben erwähnte Tabelle mit der Initialisierungszeit.

[0048] Der Zertifizierungsserver **13** ist dazu eingerichtet, eine Zertifikaterteilungsanforderung zu empfangen und das angeforderte Zertifikat für einen öffentlichen Schlüssel auszugeben, vorausgesetzt, daß das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit (d. h. dem Zeitpunkt, an dem die Zertifikaterteilungsanforderung beim Zertifizierungsserver **13** erhalten wurde) innerhalb eines maximalen Zeitintervalls **14** liegt, das beispielsweise im Zertifizierungsserver **13** gespeichert ist. Das Zeitintervall **14** kann durch einen autorisierten Benutzer konfiguriert werden, z. B. einen Netzwerk-Operator. Zur Entscheidung, ob das Zertifikat erteilt wird, greift der Zertifizierungsserver auf die Managementinformations-Datenbasis zu und liest die gespeicherte Initialisierungszeit für den gemanagten Knoten **4** aus, von dem die Zertifikaterteilungsanforderung erhalten wurde.

[0049] Um sicherzustellen, daß die Uhr des Managementservers (welche die in der Managementinformations-Datenbasis **12** gespeicherte Initialisierungszeit definiert hat) und die Uhr des Zertifizierungsservers (welche die Anforderungszeit definiert) synchronisiert sind, ist ein Zeitserver **15** vorgesehen, der beide Server **8**, **13** mit Zeitinformation versorgt, beispielsweise auf der Grundlage des Simple Network Time Protocol. Bei anderen Ausführungsformen liegen der Managementserver **8** und der Zertifizierungsserver **13** auf einer Maschine und greifen auf die gleiche Computeruhr zu; die Relativzeit zwischen

der Initialisierungszeit und der Anforderungszeit ist bei diesen Ausführungsformen richtig, ohne Synchronisierung oder ähnliches.

[0050] Das Verzeichnis **16** für öffentliche Schlüssel und Zertifikate speichert den öffentlichen Schlüssel und das zugehörige Zertifikat für den öffentlichen Schlüssel aller Netzwerkelemente und liefert auf Anforderung den öffentlichen Schlüssel und das Zertifikat eines bestimmten Netzwerkelements. Das Verzeichnis **16** für öffentliche Schlüssel und Zertifikate kann zusätzlich zu den oder statt der Speicher **6**, **7**, **10**, **11** für öffentliche Schlüssel und Zertifikate in den einzelnen Managementelementen **2**, **8** verwendet werden. Wie oben bereits erwähnt wurde, veranschaulicht Fig. 1 das IT-Netzwerk **1** in einem Zustand, in dem der Agent des dritten gemanagten Knotens **2c** noch nicht installiert ist. Der Managementserver **8** sendet gerade an den dritten gemanagten Knoten **2c** über das Netzwerk **1** ein mit "**17**" bezeichnetes Agentensoftwarepaket. Der zweite Knoten **2b** ist in einem Zustand, in dem der Agent **4b** bereits installiert ist. Der Agent **4b** hat bereits einen privaten Schlüssel **5** und einen öffentlichen Schlüssel **6** erzeugt, und hat automatisch eine, in Fig. 1 mit "**18**" bezeichnete Zertifikaterteilungsanforderung an den Zertifizierungsserver **13** gesendet. Auf den Empfang dieser Anforderung **18** hin verifiziert der Zertifizierungsserver **13** daß, wie oben beschrieben, das Initialisierungs-Anforderungszeitintervall innerhalb des maximalen Zeitintervalls **14** liegt. Wenn das Ergebnis der Verifizierung positiv ist, gibt er ein Zertifikat für den öffentlichen Schlüssel aus (in Fig. 1 mit gestrichelten Linien gezeigt und mit "**7b**" bezeichnet) und sendet es an den anfordernden Knoten **2b** zurück.

[0051] Fig. 2 ist ein Zeitdiagramm, das die Installation eines Agenten **4** (Bezugszeichen beziehen sich auf Fig. 1) und die Erteilung eines Zertifikats **7** für einen öffentlichen Schlüssel an den zum Agenten **4** gehörenden gemanagten Knoten **2** veranschaulicht. Fig. 2 veranschaulicht eine Ausführungsform, bei welcher der Managementserver und der Zertifizierungsserver auf derselben Maschine laufen. Zu einem Zeitpunkt t_1 sendet der kombinierte Management- und Zertifizierungsserver **8**, **13** ein Agentensoftwarepaket **17** an den Agenten **4** auf dem Knoten **2** mit dem Knoten-ID i. Das Agentensoftwarepaket **17** umfaßt einen vorläufigen geheimen Schlüssel. Zur gleichen Zeit läßt der kombinierte Management- und Zertifizierungsserver **8**, **13** die Initialisierungszeit (d. h. den Zeitpunkt t_1) als ein Attribut des gemanagten Knotens i in der Managementinformations-Datenbasis **12** speichern. Das Agentensoftwarepaket **17** wird zum Zeitpunkt t_2 beim Knoten i empfangen. In dem Zeitintervall von t_2 bis t_3 wird eine Selbstinstallation des Agenten **4** auf dem Knoten i durchgeführt. Sie umfaßt die Erzeugung eines privaten Schlüssels **5** und eines öffentlichen Schlüssels **6**, die zum Knoten i gehören. Beide Schlüssel **5**, **6** werden im Knoten i gespeichert. Der private Schlüssel **5** ist unsichtbar gespeichert. Der öffentliche Schlüssel **6** kann auch in dem Verzeichnis **16** für öffentliche Schlüssel und Zertifikate gespeichert werden. Am Ende der Selbstinstallierungsprozedur, zum Zeitpunkt t_3, sendet der Agent automatisch eine Anforderung zur Erteilung eines Zertifikats für den öffentlichen Schlüssel an den Zertifizierungsserver **13**. Die Anforderung, welche den öffentlichen Schlüssel **6** des Knoten enthält, ist mit dem vorläufigen geheimen Schlüssel verschlüsselt. Zum Zeitpunkt t_4 wird die Zertifikaterteilungsanforderung bei dem kombinierten Management- und Zertifizierungsserver **8**, **13** empfangen. Der Zeitpunkt t_4 ist die "Anforderungszeit". Sie wird vorübergehend in den Management- und Zertifizierungsserver **8**, **13** gespeichert. Zum Zeitpunkt t_5 fordert der Management- und Zertifizierungsserver **8**, **13** die Initialisierungszeit von der Managementinformations-Datenbasis **12** an, welche diese Anforderung bei t_6 empfängt und die Initialisierungszeit bei t_7 zurücksendet. Sie wird zum Zeitpunkt t_8 empfangen. Im Zeitintervall von t_8 bis t_9 verifiziert der Management- und Zertifizierungsserver **8**, **13**, daß das Zeitintervall zwischen der Anforderungszeit und der Initialisierungszeit nicht größer als das maximale Zeitintervall **14** ist. Wenn das Ergebnis dieses Verifizierungsschritts positiv ist, gibt der Management- und Zertifizierungsserver **8**, **13** das angeforderte Zertifikat **7** für den öffentlichen Schlüssel aus und sendet es an den anfordernden Knoten i. Das Zertifikat **7** ist mit dem vorläufigen geheimen Schlüssel verschlüsselt. Der Knoten i empfängt das Zertifikat **7** bei t_{10} und speichert es.

[0052] Fig. 3 ist ein Flußdiagramm des Agenteninitialisierungs- und Zertifikaterteilungs-Prozesses, der mit dem kombinierten Management- und Zertifizierungsserver von Fig. 2 ausgeführt wird. In Schritt S1 wird die selbstinstallierende Agentensoftware zu einer "Initialisierungszeit" an einen Knoten übersendet. Wie im Zusammenhang mit Fig. 2 beschrieben wurde, wird daraufhin ein Agent in dem Knoten installiert; er erzeugt automatisch am Ende der Installierungsprozedur eine Anforderung eines Zertifikats für den öffentlichen Schlüssel. In Schritt S2 erhält der Management- und Zertifizierungsserver zu einer "Anforderungszeit" die automatisch erzeugte Anforderung des Zertifikats. In Schritt S3 prüft der Server, ob das Zeitintervall von der Anforderungszeit zur Initialisierungszeit kleiner oder gleich dem maximalen Zeitintervall ist. Wenn die Antwort negativ ist, wird die angeforderte automatische Erteilung des Zertifikats in Schritt S4 verweigert. Wenn die Antwort positiv ist, wird das angeforderte Zertifikat in Schritt S5 automatisch erteilt und an den Knoten gesendet.

[0053] Fig. 4 ist eine vereinfachte (für Menschen lesbare) Repräsentation eines Zertifikats für einen öffentlichen Schlüssel, das einem gemanagten Knoten durch den Zertifizierungsserver erteilt wurde. Während sich normalerweise ein Zertifikat für einen öf-

fentlichen Schlüssel auf eine Person oder ein Unternehmen bezieht, binden die Zertifikate der Ausführungsformen den öffentlichen Schlüssel an ein bestimmtes Netzwerkelement, das durch den Identifikator des Netzwerkelements gekennzeichnet ist. Der Identifikator kennzeichnet das Netzwerkelement selbst; er ist unabhängig von der IP-Adresse des Netzwerkelements oder ähnlichem. Der öffentliche Schlüssel und der Netzwerkelementidentifikator bilden einen ersten Teil des Zertifikats, genannt Zertifikatkörper **21**. Der zweite Teil des Zertifikats ist eine Signatur **22**. Sie ist beispielsweise ein Hash-Wert des Zertifikatkörpers **21**, signiert mit dem privaten Schlüssel des Zertifizierungsservers. Tatsächlich hat das in den Ausführungsformen verwendete Zertifikat nicht die vereinfachte, für Menschen lesbare Form der Fig. 4, sondern es ist in einer maschinenlesbaren Form codiert, beispielsweise gemäß einer X.509 genannten Norm für Zertifikate, die im Internet breite Verwendung findet (siehe Tanenbaum, S. 767–768).

[0054] Fig. 5 ist ein Flußdiagramm einer Managementkommunikation, an deren Beginn eine sichere Sitzung eingerichtet wird. In Schritt T1 wird eine Anforderung einer Managementkommunikation von einem ersten Managementelement (z. B. einem gemanagten Knoten) an ein zweites Managementelement (z. B. den Managementserver oder einen anderen gemanagten Knoten) gesendet. In Schritt T2 wird eine gegenseitige Authentisierung des ersten und zweiten Managementelements unter Verwendung deren öffentlicher Schlüssel durchgeführt, beispielsweise gemäß dem SSL-Subprotokoll zum Errichten einer sicheren Verbindung. Die gegenseitige Authentisierung umfaßt ein Verifizieren der Authentizität der öffentlichen Schlüssel durch die diesen zugeordneten Zertifikate für den jeweiligen öffentlichen Schlüssel, wie unten im Zusammenhang mit Fig. 7 genauer erklärt wird. Während der Authentisierungsprozedur wird ein geheimer Sitzungsschlüssel zwischen dem ersten und zweiten Managementelement ausgetauscht. Er wird mit einem der öffentlichen Schlüsse der Managementelemente verschlüsselt und mit dem komplementären privaten Schlüssel entschlüsselt. Folglich ist der geheime Schlüssel nur dem ersten und zweiten Managementelement bekannt. In Schritt T3 ist die sichere Sitzung errichtet. Während der sicheren Sitzung ausgetauschte Managementnachrichten werden mit dem geheimen Sitzungsschlüssel verschlüsselt, beispielsweise gemäß dem SSL-Subprotokoll für Datenübertragung unter Verwendung der sicheren Verbindung.

[0055] Fig. 6 ist ein Flußdiagramm einer anderen Ausführungsform von authentisierter Managementkommunikation unter Verwendung digitaler Signaturen. In Schritt U1 erzeugt ein erstes Managementelement einen Hash-Wert einer Managementnachricht, und verschlüsselt ihn mit seinem privaten Schlüssel. Das erste Managementelement sendet die Nachricht zusammen mit dem verschlüsselten Hash-Wert an das zweite Managementelement. In Schritt U2 verifiziert das zweite Managementelement die Authentizität des öffentlichen Schlüssels des ersten Elements mit Hilfe des zugehörigen Zertifikats für den öffentlichen Schlüssel. Wenn die Authentizität des öffentlichen Schlüssels verifiziert ist, entschlüsselt es den Hash-Wert mit dem öffentlichen Schlüssel des ersten Elements. In Schritt U3 erzeugt das zweite Managementelement einen (weiteren) Hash-Wert der Nachricht und vergleicht ihn mit dem entschlüsselten Hash-Wert. Wenn beide Hash-Werte gleich sind, so ist damit verifiziert, daß die Nachricht von dem ersten Managementelement stammt.

[0056] Fig. 7 ist ein Flußdiagramm einer Prozedur zur Verifizierung eines öffentlichen Schlüssels auf der Grundlage des zu dem öffentlichen Schlüssel gehörenden Zertifikats, beispielsweise des in Fig. 4 gezeigten Zertifikats **7**. In Schritt V1 wird die Signatur **22** (Fig. 4) des Zertifikats mit dem öffentlichen Schlüssel des Zertifizierungsservers entschlüsselt. Es wird angenommen, daß der öffentliche Schlüssel des Zertifizierungsserver ein "Vertrauensanker" ist, der nicht verifiziert zu werden braucht. Beispielsweise kann der öffentliche Schlüssel des Zertifizierungsservers unveränderbar in jedem Managementelement gespeichert sein. In Schritt V2 wird ein Hash-Wert des Zertifikatkörpers **21** (Fig. 4) des Zertifikats erzeugt. In Schritt V3 wird geprüft, ob dieser Hash-Wert gleich der entschlüsselten Signatur ist. Wenn die Antwort positiv ist, so ist hierdurch verifiziert, daß der öffentliche Schlüssel authentisch ist (Schritt V4). Wenn die Antwort jedoch negativ ist, so ist der öffentliche Schlüssel nicht authentisch (Schritt V5).

[0057] Fig. 8 zeigt eine schematische Darstellung eines beispielhaften Computersystems **100** des IT-Netzwerks, in dem ein Satz von Instruktionen ausgeführt werden kann, um das Computersystem zum Ausführen jeglicher der hier beschriebenen Verfahren zu veranlassen. Das Computersystem **100** kann beispielsweise einer der gemanagten Knoten **2**, der Managementserver **8** oder der Zertifizierungsserver **13** von Fig. 1 sein. Es umfaßt einen Prozessor **101**, einen Speicher, beispielsweise einen Hauptspeicher **102**, und eine Netzwerkschnittstelleneinrichtung **103**, durch die es mit dem IT-Netzwerk gekoppelt ist. Die Komponenten des Computersystems **100** kommunizieren über einen Bus **104** miteinander. Weitere fakultative Speicherkomponenten können vorhanden sein, beispielsweise ein statischer Speicher **105** und eine Disketteneinheit **106** mit einem maschinenlesbaren Medium **107**. Das Computersystem **100** kann fakultativ auch ein Ausgabegerät **108**, z. B. einen Bildschirm, und ein Eingabegerät **109**, z. B. eine alphanumerische Tastatur und/oder eine Cursor-Steuereinrichtung umfassen.

[0058] Ein Satz von Instruktionen (d. h. Software)

110, der eines oder alle der oben beschriebenen Verfahren verkörpert, liegt vollständig oder wenigstens teilweise im Prozessor **101** und/oder dem Hauptspeicher **102**. Die Software **110** ist auch als vollständig oder wenigstens teilweise im statischen Speicher **105** und auf dem maschinenlesbaren Medium **107** gespeichert dargestellt. Sie kann außerdem über die Netzwerkschnittstelleneinrichtung **103** über das IT-Netzwerk versendet oder empfangen werden. Die Software **110** kann über mehrere Prozessoren oder Computer verteilt sein, beispielsweise über mehrere gemanagte Knoten, den Managementserver und den Zertifizierungsserver.

[0059] Die offenbarten Ausführungsformen stellen somit verbesserte Verfahren und Erzeugnisse bereit, welche eine Verringerung der Menge nötiger menschlicher Interaktion im Netzwerkmanagement ermöglichen, ohne ein unakzeptables Sicherheitsloch zu erzeugen.

Patentansprüche

1. Verfahren zum Einrichten eines Managementagenten (**4**) in einem Knoten (**2**) eines IT-Netzwerks (**1**), in dem gemanagte Knoten (**2**) in der Management-Kommunikation authentisiert werden, wobei die Authentisierung auf Kryptographie mit öffentlichen Schlüsseln beruht, umfassend:
Einleitung einer automatisierten Installation eines Management-Agenten (**4**) auf einem Knoten (**2**) durch einen Management-Server (**8**) durch Übertragung der zu installierenden Agentensoftware und/oder einer Installationsanforderung vom Management-Server (**8**) auf den Knoten (**2**), wobei dies eine sog. Initialisierungszeit definiert;
Automatisierte Installation des Agenten (**4**) auf dem Knoten (**2**);
Anforderung der Erteilung eines Zertifikats (**7**) für einen öffentlichen Schlüssel durch den Agenten (**4**) bei einem Zertifizierungs-Server (**13**), wobei dies eine sog. Anforderungszeit definiert;
Automatische Erteilung des angeforderten Zertifikats (**7**) durch den Zertifzierungs-Server (**13**), falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit in einem maximalen Zeitintervall für automatische Zertifikaterteilung liegt, oder Verweigerung der automatischen Erteilung des angeforderten Zertifikats (**7**) durch den Zertifizierungs-Server (**13**), falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit größer als das maximale Zeitintervall ist;
wobei das maximale Zeitintervall so gewählt wird, daß es nur etwas länger als die Zeit ist, die typischerweise zur Übertragung der Agentensoftware und zur automatischen Installation des Agenten benötigt wird, so daß ein Sicherheitsloch aufgrund der Möglichkeit einer Zertifikatserlangung nur während der Initialisierungsphase, d. h. der typischerweise zur Übertragung der Agentensoftware und zur automatischen Installation des Agenten benötigten Zeit, besteht.

2. Verfahren nach Anspruch 1, wobei das maximale Zeitintervall für automatische Zertifikaterteilung von einem Administrator oder Operator des IT-Netzwerks (**1**) konfigurierbar ist.

3. Verfahren nach einem der Ansprüche 1 oder 2, wobei die Initialisierungszeit als eine Eigenschaft des gemanagten Knotens (**2**) gespeichert wird.

4. Verfahren nach Anspruch 3, wobei die gespeicherte Initialisierungszeit verwendet wird, um zu bestimmen, ob das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit in dem maximalen Zeitintervall für automatische Zertifikaterteilung liegt.

5. Verfahren nach einem der Ansprüche 1 bis 4, wobei die Anforderungszeit der Zeitpunkt des Empfangs der Anforderung des gemanagten Knotens zur Erteilung eines Zertifikats (**7**) bei dem Zertifizierungsserver (**13**) ist.

6. Verfahren nach einem der Ansprüche 1 bis 5, wobei die Initialisierungszeit und die Anforderungszeit von verschiedenen Servern oder gemanagten Knoten (**2**) definiert werden, die synchronisierte Uhren haben.

7. Verfahren nach einem der Ansprüche 1 bis 5, wobei die Initialisierungszeit und die Anforderungszeit durch denselben Server oder gemanagten Knoten definiert werden.

8. Verfahren nach einem der Ansprüche 1 bis 7, wobei die Anforderung (**18**) zur Erteilung eines Zertifikats (**7**) bei oder nach der Installation des Managementagenten (**4**) automatisch erzeugt wird.

9. Verfahren nach einem der Ansprüche 1 bis 8, wobei, nach der Erteilung des Zertifikats (**7**) an den gemanagten Knoten (**2**), der gemanagte Knoten (**2**) in der Managementkommunikation mit einem Managementserver (**8**) oder einem anderen gemanagten Knoten (**2**) authentisiert wird, und zwar unter Verwendung des öffentlichen Schlüssels (**6**) und des Zertifikats (**7**) für den öffentlichen Schlüssel des gemanagten Knotens und eines privaten Schlüssels (**5**) des gemanagten Knotens (**2**), der zu dessen öffentlichen Schlüssel (**6**) gehört.

10. Verfahren nach Anspruch 9, wobei auch der Managementserver (**8**) oder der andere gemanagte Knoten (**2**) authentisiert wird, und zwar unter Verwendung eines öffentlichen Schlüssels (**10**, **6**) des Managementservers (**8**) oder des anderen gemanagten Knotens (**2**).

11. IT-Netzwerk-Managementsystem, umfassend:
einen Management-Server (**8**) und einen Zertifizierungsserver (**13**),
wobei der Management-Server (**8**) dazu eingerichtet ist, die Installierung eines Managementagenten (**4**) auf einem gemanagten Knoten (**2**) durch Übertragung der zu installierenden Agentensoftware und/oder einer Installationsanforderung vom Management-Server (**8**) auf den Knoten (**2**) zu einer Initialisierungszeit zu veranlassen;
wobei der Agent (**4**) dazu eingerichtet ist, bei dem Zertifizierungsserver (**13**) zu einer Anforderungszeit die Erteilung eines Zertifikats (**7**) für einen öffentlichen Schlüssel anzufordern; und
wobei der Zertifizierungsserver (**13**) dazu eingerichtet ist, das angeforderte Zertifikat (**7**) automatisch zu erteilen, falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit in einem maximalen Zeitintervall für automatische Zertifikaterteilung liegt, oder die automatische Erteilung des angeforderten Zertifikats zu verweigern, falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit größer als das maximale Zeitintervall ist,
wobei das maximale Zeitintervall so gewählt ist, daß es nur etwas länger als die Zeit ist, die typischerweise zur Übertragung der Agentensoftware und zur automatischen Installation des Agenten benötigt wird, so daß ein Sicherheitsloch aufgrund der Möglichkeit einer Zertifikatserlangung nur während der Initialisierungsphase, d. h. der typischerweise zur Übertragung der Agentensoftware und zur automatischen Installation des Agenten benötigten Zeit, besteht.

12. Computerprogramm mit Programmcode (**110**), der bei Ausführung in einem gemanagten IT-Netzwerk (**1**), in dem gemanagte Knoten (**2**) in der Managementkommunikation authentisiert werden, wobei die Authentisierung auf Kryptographie mit öffentlichem Schlüssel beruht, zum Durchführen eines Verfahrens zum Einrichten eines Managementagenten (**4**) in einem Knoten (**2**) dient, wobei der Programmcode (**110**) dazu eingerichtet ist:
eine automatische Installation eines Management-Agenten (**4**) auf einem Knoten (**2**) durch Übertragung der zu installierenden Agentensoftware und/oder einer Installationsanforderung von einem Management-Server (**8**) auf den Knoten (**2**) einzuleiten, wobei dies eine sog. Initialisierungszeit definiert;
den Agenten (**4**) auf dem Knoten (**2**) automatisch zu installieren;
durch den Agenten (**4**) bei einem Zertifizierungsserver (**13**) die Erteilung eines Zertifikats (**7**) für einen öffentlichen Schlüssel anzufordern, wobei dies eine sog. Anforderungszeit definiert;
durch den Zertifizierungsserver (**13**) das angeforderte Zertifikat (**7**) automatisch zu erteilen, falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit in einem maximalen Zeitintervall für automatische Zertifikaterteilung liegt, oder die automatische Erteilung des angeforderten Zertifikats (**7**) zu verweigern, falls das Zeitintervall zwischen der Initialisierungszeit und der Anforderungszeit größer als das maximale Zeitintervall ist;
wobei das maximale Zeitintervall so gewählt wird, daß es nur etwas länger als die Zeit ist, die typischerweise zur Übertragung der Agentensoftware und zur automatischen Installation des Agenten benötigt wird, so daß ein Sicherheitsloch aufgrund der Möglichkeit einer Zertifikatserlangung nur während der Initialisierungsphase, d. h. der typischerweise zur Übertragung der Agentensoftware und zur automatischen Installation des Agenten benötigten Zeit, besteht.

Es folgen 8 Blatt Zeichnungen

Anhängende Zeichnungen

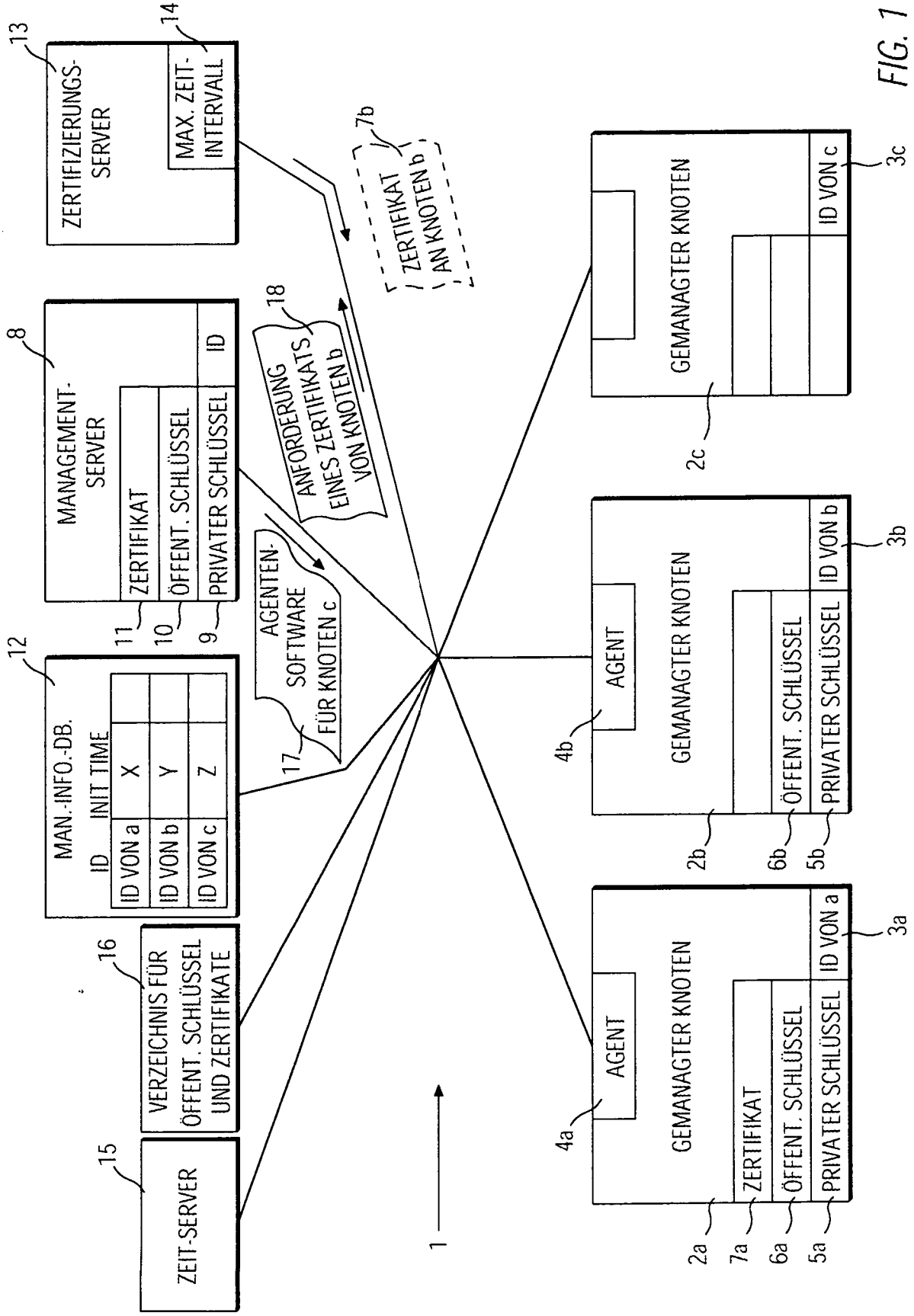


FIG. 1

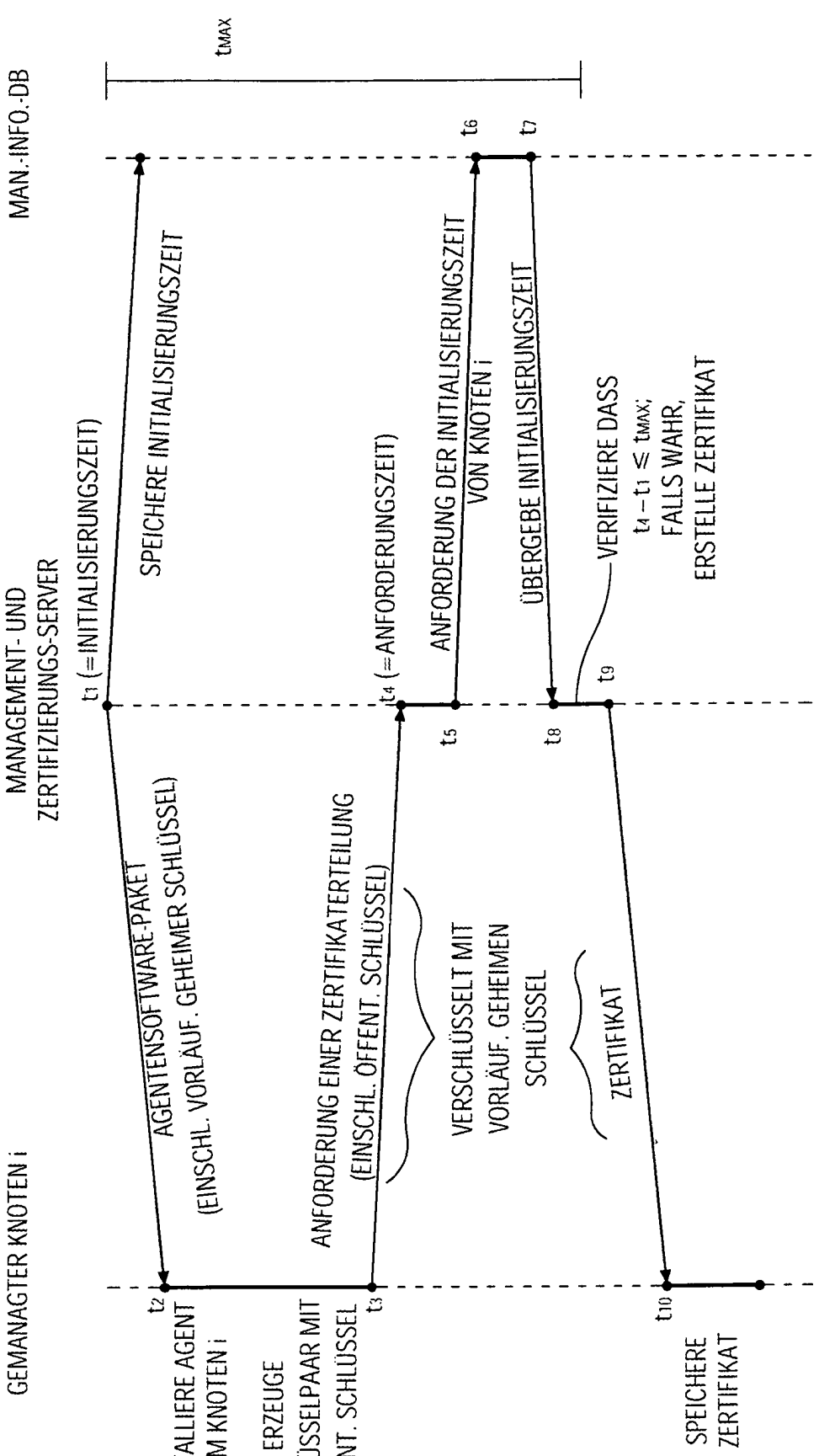

GEMANAGTER KNOTEN i
MANAGEMENT- UND ZERTIFIZIERUNGS-SERVER
MAN.-INFO.-DB
t_{MAX}
t_1 (=INITIALISIERUNGSZEIT)
SPEICHERE INITIALISIERUNGSZEIT
AGENTENSOFTWARE-PAKET (EINSCHL. VORLÄUF. GEHEIMER SCHLÜSSEL)
t_2
INSTALLIERE AGENT IM KNOTEN i
ERZEUGE SCHLÜSSELPAAR MIT ÖFFENT. SCHLÜSSEL
t_3
ANFORDERUNG EINER ZERTIFIKATERTEILUNG (EINSCHL. ÖFFENT. SCHLÜSSEL)
t_4 (=ANFORDERUNGSZEIT)
t_5
VERSCHLÜSSELT MIT VORLÄUF. GEHEIMEN SCHLÜSSEL
ANFORDERUNG DER INITIALISIERUNGSZEIT VON KNOTEN i
t_6
ÜBERGEBE INITIALISIERUNGSZEIT
t_7
t_8
ZERTIFIKAT
VERIFIZIERE DASS $t_4 - t_1 \leq t_{MAX}$; FALLS WAHR, ERSTELLE ZERTIFIKAT
t_9
t_{10}
SPEICHERE ZERTIFIKAT

FIG. 2

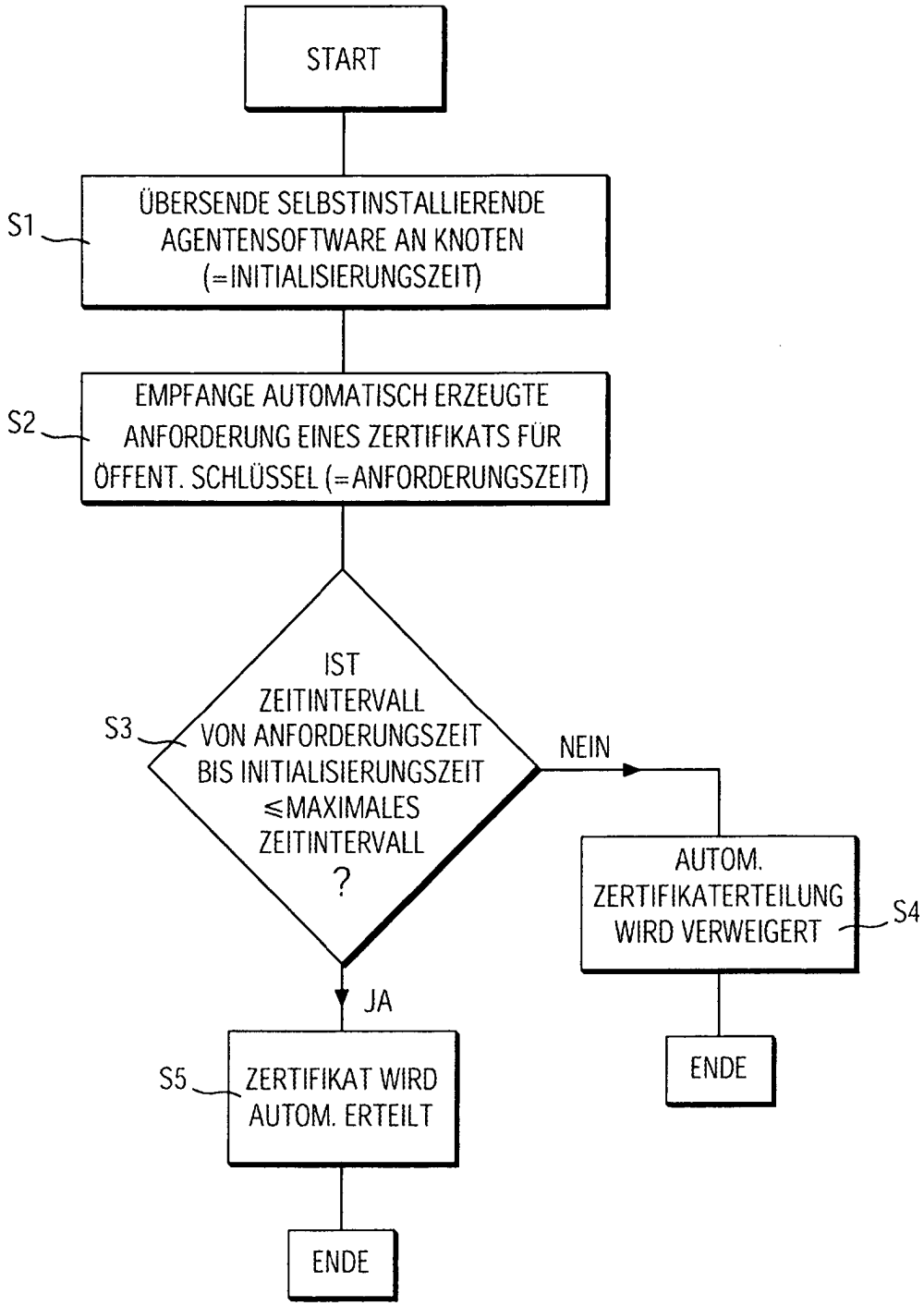

START
S1
ÜBERSENDE SELBSTINSTALLIERENDE AGENTENSOFTWARE AN KNOTEN (=INITIALISIERUNGSZEIT)
S2
EMPFANGE AUTOMATISCH ERZEUGTE ANFORDERUNG EINES ZERTIFIKATS FÜR ÖFFENT. SCHLÜSSEL (=ANFORDERUNGSZEIT)
S3
IST ZEITINTERVALL VON ANFORDERUNGSZEIT BIS INITIALISIERUNGSZEIT ≤MAXIMALES ZEITINTERVALL ?
NEIN
AUTOM. ZERTIFIKATERTEILUNG WIRD VERWEIGERT
S4
ENDE
JA
S5
ZERTIFIKAT WIRD AUTOM. ERTEILT
ENDE

FIG. 3

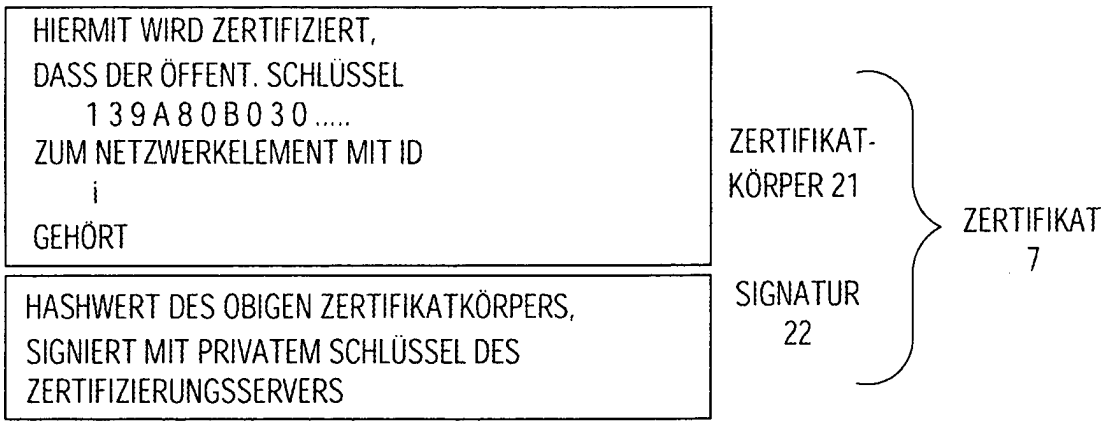

HIERMIT WIRD ZERTIFIZIERT,
DASS DER ÖFFENT. SCHLÜSSEL
1 3 9 A 8 0 B 0 3 0 .....
ZUM NETZWERKELEMENT MIT ID
i
GEHÖRT
HASHWERT DES OBIGEN ZERTIFIKATKÖRPERS,
SIGNIERT MIT PRIVATEM SCHLÜSSEL DES
ZERTIFIZIERUNGSSERVERS
ZERTIFIKAT-
KÖRPER 21
SIGNATUR
22
ZERTIFIKAT
7

FIG. 4

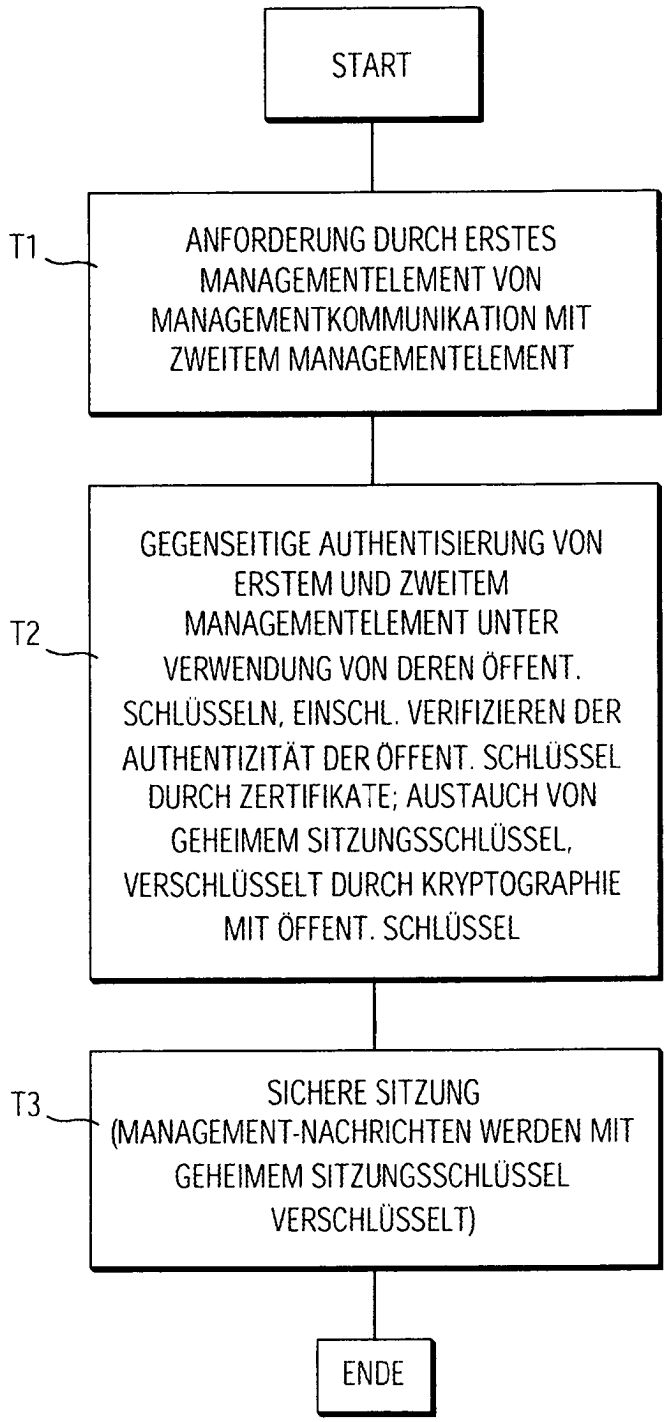

START
T1
ANFORDERUNG DURCH ERSTES MANAGEMENTELEMENT VON MANAGEMENTKOMMUNIKATION MIT ZWEITEM MANAGEMENTELEMENT
T2
GEGENSEITIGE AUTHENTISIERUNG VON ERSTEM UND ZWEITEM MANAGEMENTELEMENT UNTER VERWENDUNG VON DEREN ÖFFENT. SCHLÜSSELN, EINSCHL. VERIFIZIEREN DER AUTHENTIZITÄT DER ÖFFENT. SCHLÜSSEL DURCH ZERTIFIKATE; AUSTAUCH VON GEHEIMEM SITZUNGSSCHLÜSSEL, VERSCHLÜSSELT DURCH KRYPTOGRAPHIE MIT ÖFFENT. SCHLÜSSEL
T3
SICHERE SITZUNG (MANAGEMENT-NACHRICHTEN WERDEN MIT GEHEIMEM SITZUNGSSCHLÜSSEL VERSCHLÜSSELT)
ENDE

FIG. 5

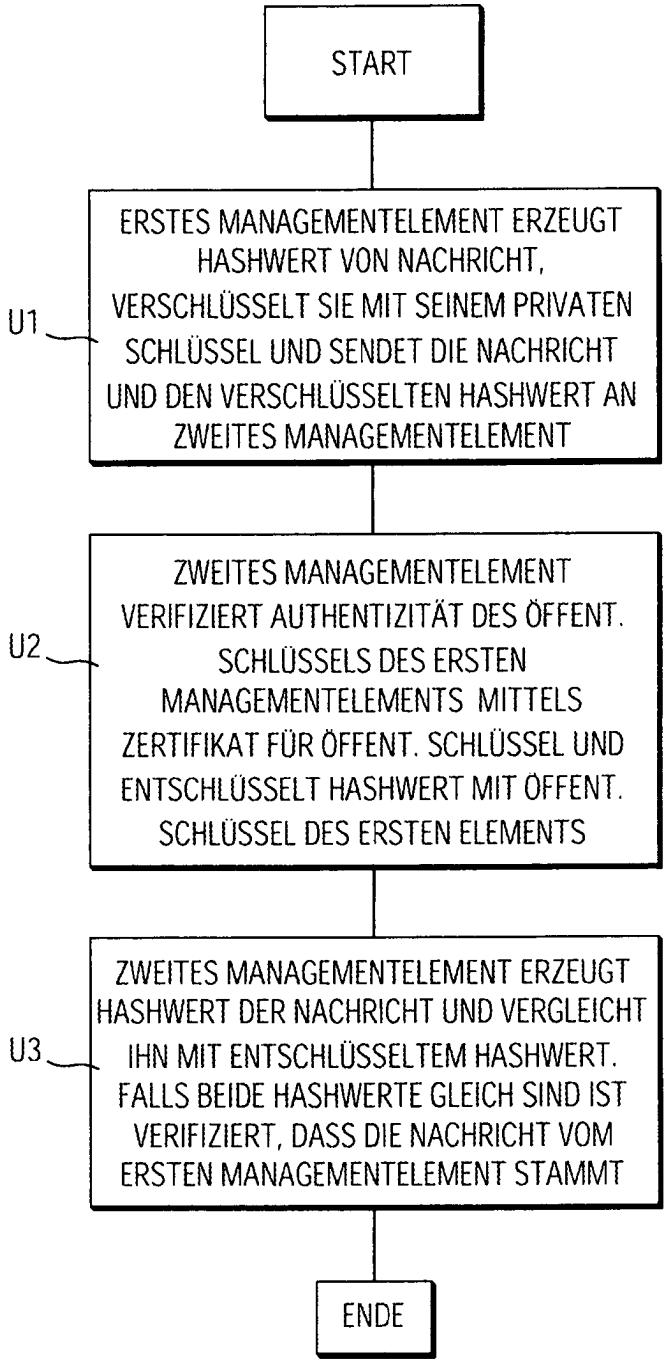

START
ERSTES MANAGEMENTELEMENT ERZEUGT HASHWERT VON NACHRICHT, VERSCHLÜSSELT SIE MIT SEINEM PRIVATEN SCHLÜSSEL UND SENDET DIE NACHRICHT UND DEN VERSCHLÜSSELTEN HASHWERT AN ZWEITES MANAGEMENTELEMENT
U1
ZWEITES MANAGEMENTELEMENT VERIFIZIERT AUTHENTIZITÄT DES ÖFFENT. SCHLÜSSELS DES ERSTEN MANAGEMENTELEMENTS MITTELS ZERTIFIKAT FÜR ÖFFENT. SCHLÜSSEL UND ENTSCHLÜSSELT HASHWERT MIT ÖFFENT. SCHLÜSSEL DES ERSTEN ELEMENTS
U2
ZWEITES MANAGEMENTELEMENT ERZEUGT HASHWERT DER NACHRICHT UND VERGLEICHT IHN MIT ENTSCHLÜSSELTEM HASHWERT. FALLS BEIDE HASHWERTE GLEICH SIND IST VERIFIZIERT, DASS DIE NACHRICHT VOM ERSTEN MANAGEMENTELEMENT STAMMT
U3
ENDE

FIG. 6

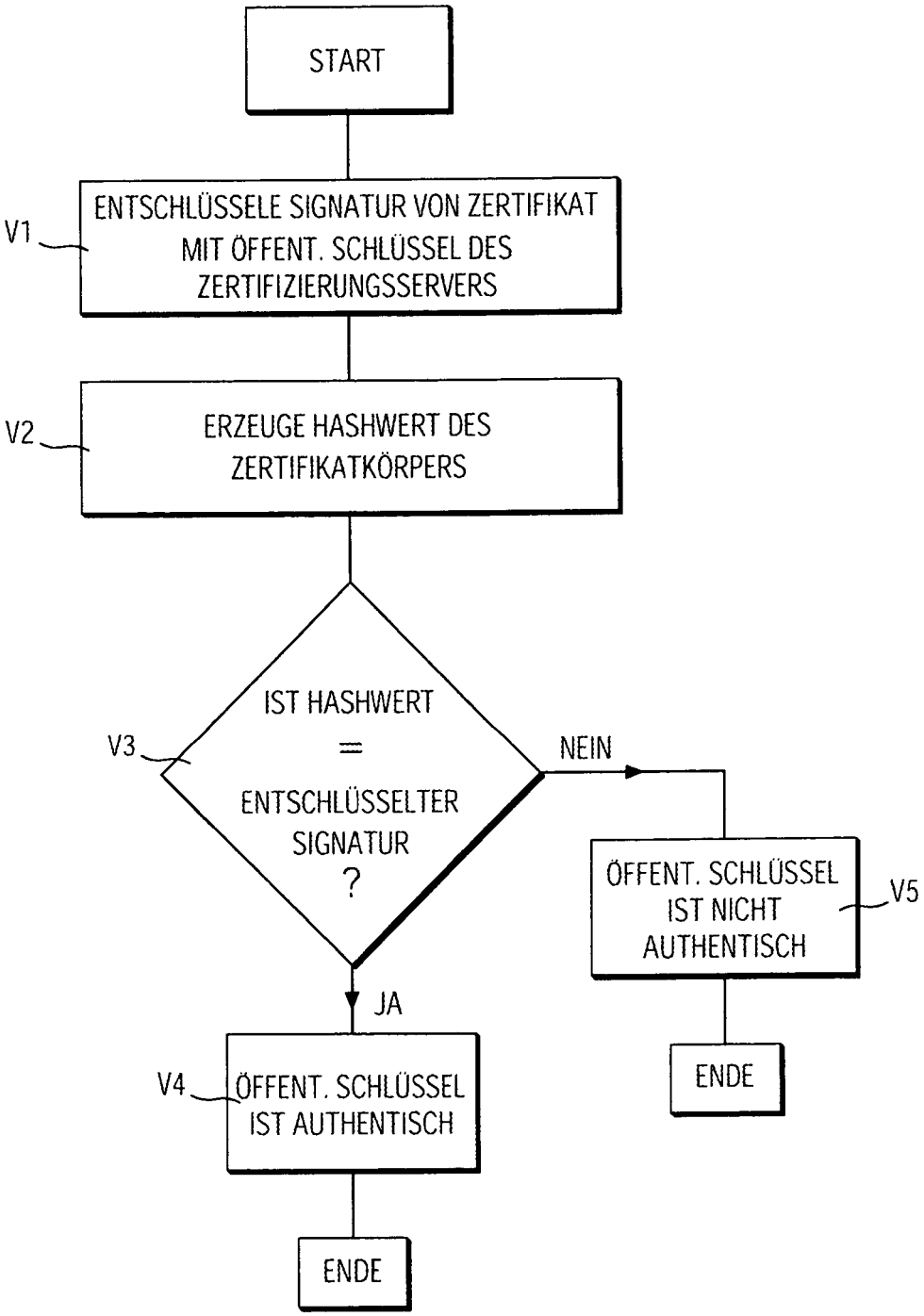

START
ENTSCHLÜSSELE SIGNATUR VON ZERTIFIKAT MIT ÖFFENT. SCHLÜSSEL DES ZERTIFIZIERUNGSSERVERS
V1
ERZEUGE HASHWERT DES ZERTIFIKATKÖRPERS
V2
IST HASHWERT = ENTSCHLÜSSELTER SIGNATUR ?
V3
NEIN
JA
ÖFFENT. SCHLÜSSEL IST NICHT AUTHENTISCH
V5
ENDE
ÖFFENT. SCHLÜSSEL IST AUTHENTISCH
V4
ENDE

FIG. 7

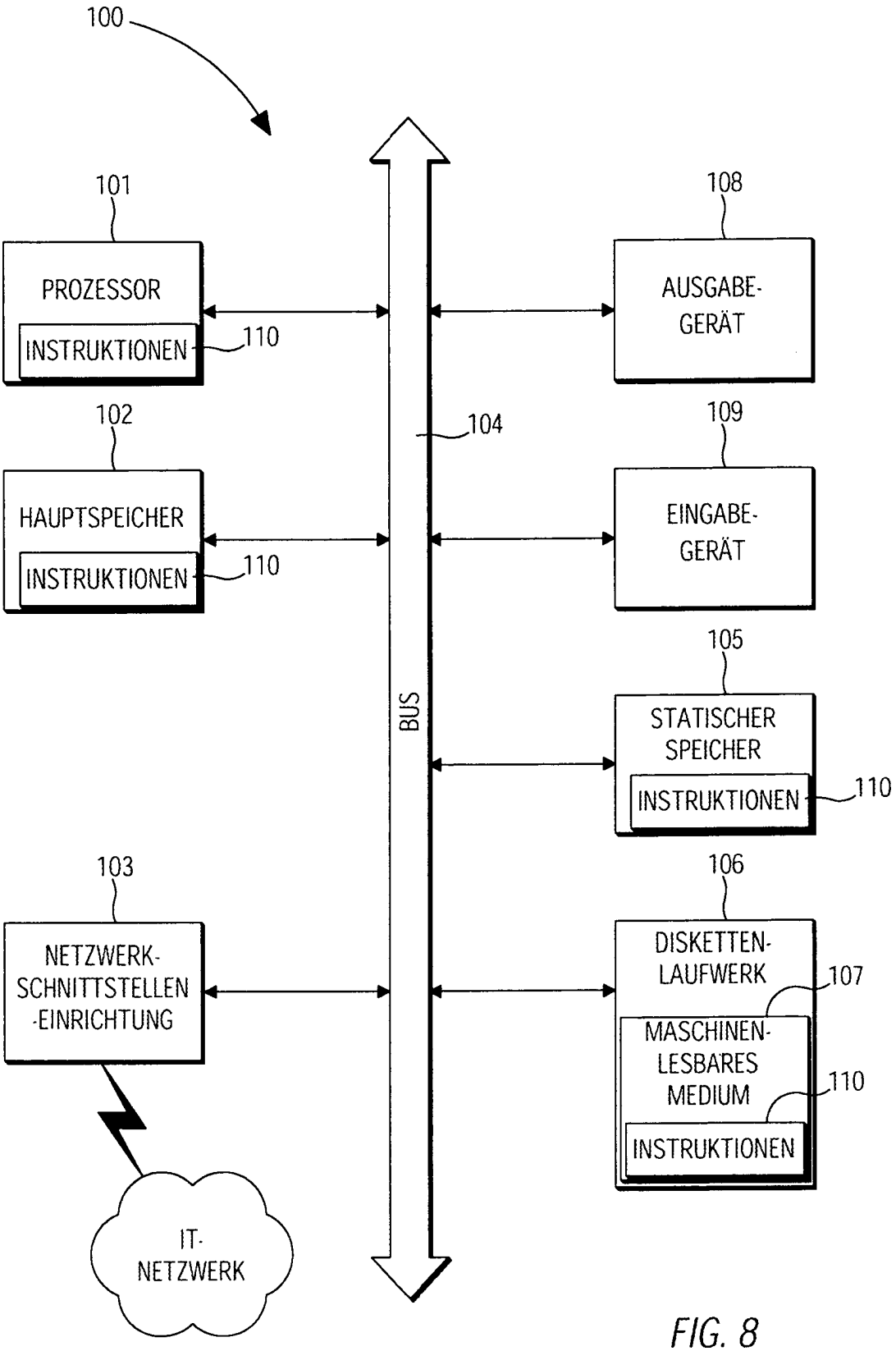

100
101
PROZESSOR
INSTRUKTIONEN
110
102
HAUPTSPEICHER
INSTRUKTIONEN
110
103
NETZWERK-
SCHNITTSTELLEN
-EINRICHTUNG
IT-
NETZWERK
104
BUS
108
AUSGABE-
GERÄT
109
EINGABE-
GERÄT
105
STATISCHER
SPEICHER
INSTRUKTIONEN
110
106
DISKETTEN-
LAUFWERK
107
MASCHINEN-
LESBARES
MEDIUM
110
INSTRUKTIONEN

FIG. 8