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

(10) **DE 699 23 503 T2** 2006.02.16

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 977 399 B1**
(21) Deutsches Aktenzeichen: **699 23 503.0**
(96) Europäisches Aktenzeichen: **99 305 924.5**
(96) Europäischer Anmeldetag: **26.07.1999**
(97) Erstveröffentlichung durch das EPA: **02.02.2000**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **02.02.2005**
(47) Veröffentlichungstag im Patentblatt: **16.02.2006**

(51) Int Cl.[8]: ***H04L 12/24*** (2006.01)
H04L 29/06 (2006.01)

(30) Unionspriorität:
124181 **28.07.1998** **US**

(84) Benannte Vertragsstaaten:
DE, FR, GB

(73) Patentinhaber:
Sun Microsystems, Inc., Palo Alto, Calif., US

(74) Vertreter:
HOFFMANN & EITLE, 81925 München

(72) Erfinder:
Chang, April S., Los Altos, US; Large, Andrew R., La Selva Beach CA 95076, US; Snyder, Alan, Palo Alto, US

(54) Bezeichnung: **Authentifizierung und Zugriffskontrolle in einem Managementterminalprogramm zur Verwaltung von Diensten in einem Computernetzwerk**

Beschreibung

HINTERGRUND DER ERFINDUNG

1. GEBIET DER ERFINDUNG

[0001] Die vorliegende Erfindung bezieht sich allgemein auf Computersoftware und Computernetzmanagement. Genauer bezieht sich die vorliegende Erfindung auf serverbasierte Managementsoftware und Softwareregistration in einem Computernetz.

2. ERÖRTERUNG DES STANDES DER TECHNIK

[0002] In den letzten Jahren sind Computernetze nicht nur in der Größe gewachsen, wie etwa der Zahl von Benutzern oder geografischer Abdeckung, sondern auch im Sinne der Typen von Diensten und Protokollen, die ein einzelnes Netz vorsehen und unterstützen kann. Viele Computernetze erlauben Endbenutzern, auf alle Typen von Diensten zuzugreifen, wie etwa Durchsuchen von Nachrichtendiensten oder Zugreifen auf das Internet, und schränken Benutzer nicht auf ein zwingendes oder erforderliches Netzkommunikationsprotokoll ein. Mit der Ausbreitung von Diensten, die in einigen Computernetze verfügbar sind, gibt es eine wachsende Belastung bei System- oder Netzadministratoren beim Managen dieser Dienste. Ein Systemadministrator muss nun typischerweise Software auf verschiedenen Servern installieren und managen, wobei jeder Server typischerweise einen oder mehr Dienste für Netzbenutzer unterbringt oder vorsieht. Abhängig von der Größe des Netzes und der Zahl von Diensten kann das tägliche Management, z.B. Installieren, Aktualisieren und Beheben von Störungen, der Software, die hinter diesen Diensten steht, eine langwierige, fehleranfällige und zeitraubende Aufgabe für einen Systemadministrator werden. Dies trifft insbesondere hinsichtlich Systemadministratoren zu, die mit dem Netz, den Servern oder der Konfiguration dieser Server nicht vertraut sind.

[0003] Die Literaturstelle SZWARC M: "Virtual private data network service in the wide area networks" COUNTDOWN TO THE NEW MILLENIUM. PHOENIX; 2.–5. DEZ. 1991, PROCEEDINGS OF THE GLOBAL TELECOMMUNICATIONS CONFERENCE. (GLOBECOM), NEW YORK, IEEE, US, Vol. 3, 2. Dezember 1991, Seiten 1033–1037, ISBN: 0-87942-697-7 legt ein Verfahren und eine Vorrichtung für einen sicheren Zugriff auf die Administration einer Vielzahl von Netzdiensten offen, die sich auf einem oder mehr Dienst-Host-Computern befinden. Der Benutzer meldet sich bei einem zentralen Dienstmanager mit USERID und Passwort an, und ihm wird dann eine Liste von Diensten präsentiert, die er verwenden und/oder administrieren kann.

[0004] In einem Computernetz mit großem Ausmaß, das typischerweise viele Typen von Diensten und Anwendungen, wie oben beschrieben, vorsieht, gibt es typischerweise mehrere oder viele Server-Maschinen, auf die Endbenutzer oder Clients zugreifen können. Die Tatsache, dass es viele Server in dem Netz gibt, ist gewöhnlich für einen typischen Endnutzer transparent, der normalerweise nicht mit der physischen Konfiguration des Netzes befasst ist. Ein Systemadministrator, der zum Managen eines Computernetzes verantwortlich ist, tut dies normalerweise von einem Server und einer Konsole, der generisch als ein Administrationsserver beschrieben wird, wie etwa ein Webserver. Fig. 1 ist ein Blockdiagramm eines Computernetzes mit vielen Servern, die für Endbenutzer zugreifbar und mit einem Administrationsserver verbunden sind, der nicht mit den automatisierten Managementfähigkeiten der vorliegenden Erfindung konfiguriert ist. Ein Computernetz **102** hat eine Administratorkonsole, gezeigt als Client **104**, die mit einem Web- oder Administratorserver **106** verbunden ist. Mit dem Webserver **106** sind viele "Dienst-"Server **108** verbunden. Aus der Sicht des Administrationsservers **106** werden Server **108** als Managementclients bezeichnet. Aus der Sicht eines Endbenutzers sind sie aber einfach Server, wobei jeder Server eine bestimmte Funktion aufweisen oder einen bestimmten Dienst vorsehen kann.

[0005] Wenn eine Aktualisierung, Installation oder ein beliebiger Typ von Wartung in der Anwendungssoftware vorgenommen wird, die sich auf einem der Server **108** befindet, oder dem Netz **102** ein neuer Server hinzugefügt wird, muss der Systemadministrator Software auf Administrationsserver **106** entsprechend modifizieren. Falls z.B. ein neues Merkmal auf einem existierenden Mail-Server installiert wird oder ein neuer Mail-Server hinzugefügt wird, muss sich der Administrator den Standort und andere Informationen des neuen Merkmals oder Servers zur Zeit der Aktualisierung aufschreiben oder merken. Der Administrator installiert eine neue Anwendung auf einem Server **110**. Diese Information, inkludierend den Standort von beliebigen Managementmodulen der neuen Anwendung, die in der Form eines einheitlichen Ressourcenlokators (Uniform Resource Locator) sein kann, muss dann in Konsole **104** eingegeben werden. Sobald in Administratorkonsole **104** manuell eingegeben, wird die Information, die benötigt wird, um die neue Software oder den Server zum managen, auf Administrationsserver **106** widergespiegelt. In dieser Stufe ist der Standort von beliebigen Managementmodulen auf Server **110** für den Systemadministrator von Administratorkonsole **104** verfügbar. Das neue Mail-Merkmal aus dem Beispiel kann durch Endbenutzer nicht gemanagt oder richtig konfiguriert werden, bis es bei dem Administrationsserver **106** "registriert" ist. Administrationsserver **106** muss wissen, wo die Managementmodule auf Managementclients **108** zu finden sind, die mit dem neuen Mail-Merkmal in Verbindung stehen, bevor Endbe-

nutzer eine Verwendung der Software beginnen können.

[0006] Dies ist ein ineffizienter Prozess für den Administrator und für Endbenutzer unbequem, die in der Erwartung gekommen sind, dass neue Anwendungen in ihren Netzen für eine Verwendung so schnell wie möglich verfügbar sind. Dieser Prozess ist auch fehleranfällig, da der Administrator manuelle oder nicht-automatisierte Aufgaben durchführen muss, wie Aufschreiben von Information über das neue Merkmal oder den Server während einer Installation, die später auf einer Administratorkonsole einzugeben ist. Dieses Problem wird verstärkt, falls es Dutzende von Servern gibt, jeder mit vielen Anwendungen (z.B. sind 30 nicht unüblich), die häufige Aktualisierungen, Korrekturen oder neue Versionen aufweisen, die auf eine zeitgerechte und akkurate Art und Weise installiert werden müssen. In diesem Typ einer Einrichtung kann Management von Netzdiensten nicht nur ineffizient, zeitraubend und fehleranfällig, sondern auch unpraktisch sein.

[0007] Ein Problem mit vorhandenen webserverbasierten Netzen, die typischerweise viele Diensthosts aufweisen, besteht in der Gestaltung und Implementierung eines Benutzerauthentifizierungsmechanismus. Ein webserverbasiertes Computernetz, oder ein beliebiger Typ eines Computernetzes, muss ein Authentifizierungsprotokoll oder Mechanismus haben um sicherzustellen, dass ein Benutzer nur diejenigen Operationen durchführen oder auf diejenigen Dateien zugreifen kann, für die der Benutzer für eine Durchführung oder einen Zugriff autorisiert ist. In dem Fall vom Managen von Diensten auf den vielen Diensthosts kann es mehr als einen Systemadministrator geben, der für eine Unterhaltung der Dienste auf diesen Hosts verantwortlich ist. Es ist möglich, dass gewissen Administratoren nicht die vollständige Autorisierung gegeben wird, alle möglichen Operationen auf dem Webserver und den Diensthosts durchzuführen, was z.B. nur einem Senior- oder "Super"-Systemadministrator gegeben werden kann. Da Managen von Diensten auf den Hosts eine Administrationsaufgabe ist, die durch eine Administrationsschnittstelle erledigt wird, ist irgendein Typ von Benutzerauthentifizierung notwendig.

[0008] Obwohl Authentifizierung für webbasierte Netze existiert, sind vorliegende Implementierungen und Gestaltungen für Benutzerautorisierung ineffizient und sich wiederholend. Die Authentifizierung, auf die hier Bezug genommen wird, ist die Verifizierung und Autorisierung von System- oder Netzadministratoren zum Managen von Diensten auf Diensthosts in einem Netz von einem Browser auf einer Administrationskonsole. Typischerweise haben jeder Dienst auf einem Diensthost und seine ein oder mehr Managementmodule unterschiedliche Authentifizierungsmechanismen und Standards. Es gibt keinen klaren Standard über ein Protokoll oder einen Prozess zum Implementieren von Authentifizierung und Zugriffssteuerung auf eine verteilte Art und Weise in einem webserverbasierten System. Ein Systemadministrator muss sich jedes Mal erneut authentifizieren, wenn sich der Administrator an einem Diensthost anmeldet, da die Diensthosts nicht miteinander in Verbindung stehen. Es kann ein Browser-Programm auf einem Client laufen, das auf einem beliebigen Typ eines Betriebssystems läuft, somit kann der Browser, der durch den Administrator verwendet wird, nicht auf einem UNIX-basierten Client sein und nicht eine bekannte UNIX-Identität haben. Da der Browser nicht eine bekannte UNIX-Identität hat, kann eine Identität nicht von einem Diensthost zu anderen Diensthosts kommuniziert werden. Somit muss ein Systemadministrator einen Authentifizierungsprozess für jeden Diensthost durchlaufen, da der Administrator nicht eine einzelne oder global anerkannte Identität hat.

[0009] Deshalb wäre es wünschenswert, Endbenutzeranwendungssoftware und Dienste, die in einem Computernetz verfügbar sind, von einem zentralen Standort zum managen, indem es eine beliebige notwendige Software zum Managen dieser Anwendungen und Dienste gibt, die automatisch in dem zentralen Standort während Installation registriert wird und von einem gut bekannten Standort zugreifbar ist. Es wäre auch wünschenswert, einen Authentifizierungsmechanismen zu haben, der eine einzelne Anmeldung für diese Funktionen innerhalb der Umgebung eines Webservers und des existierenden Systems des Servers von Benutzeridentität und Zugriffssteuerung vorsieht. Ferner wäre es wünschenswert, dies von einem zentralen Standort und durch Zuweisen einer universellen Identität für einen Benutzer, der Dienste von einem Browser in einem webserverbasierten Netz managt, zu erreichen.

ZUSAMMENFASSUNG DER ERFINDUNG

[0010] Um das Vorangehende zu erreichen, und in Übereinstimmung mit dem Zweck der vorliegenden Erfindung werden ein Verfahren und ein System und ein computerlesbares Medium zum Sichern von Zugriff zu einem Dienstmanager für die Administration von Diensten, die sich auf einem oder mehr Diensthostcomputern befinden, gemäß Ansprüchen 1, 14 bzw. 15 vorgesehen. Die Erfindung bezieht sich ferner auf ein System zum Sichern von Administration von Diensten, die sich auf einem oder mehr Diensthostcomputern befinden, von einem Administrationsservercomputer gemäß Anspruch 13.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0011] Die Erfindung, gemeinsam mit ihren weiteren Vorteilen, kann am besten durch Verweis auf die folgende Beschreibung verstanden werden, die in Verbindung mit den begleitenden Zeichnungen aufge-

nommen wird, in denen:

[0012] Fig. 1 ein Blockdiagramm eines Computernetzes mit vielen Servern ist, auf die durch Endbenutzer zugegriffen werden kann, und die mit einem Administrationsserver verbunden sind, der nicht mit den automatisierten Managementfähigkeiten der vorliegenden Erfindung konfiguriert ist;

[0013] Fig. 2 ein Blockdiagramm von serverseitigen Komponenten eines Computernetzes in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung ist;

[0014] Fig. 3 ein Flussdiagramm ist, das einen Überblick über einen Prozess zum Registrieren eines neuen Dienstes in einem Netz in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung zeigt;

[0015] Fig. 4 ein Flussdiagramm ist, das Schritt **304** von Fig. 3 zum Registrieren eines Dienstes in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung detaillierter zeigt;

[0016] Fig. 5 ein Flussdiagramm ist, das Schritt **306** von Fig. 3 in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung detaillierter zeigt;

[0017] Fig. 6a und Fig. 6b Bildschirmdrucke einer grafischen Benutzerschnittstelle sind, die auf dem Browser-Host in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung angezeigt wird;

[0018] Fig. 7 ein Bildschirmdruck einer grafischen Benutzerschnittstelle bezüglich der Zugriffssteuerung und Authentifizierung eines Benutzers des Managementkonsolenprogramms in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung ist;

[0019] Fig. 8a und Fig. 8b Flussdiagramme eines Prozesses zum Durchsetzen einer Zugriffssteuerung und Autorisierung in dem Managementsteuerprogramm in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung sind;

[0020] Fig. 9 ein Flussdiagramm ist, das Schritt **806** von Fig. 8a detaillierter zeigt;

[0021] Fig. 10 ein Blockdiagramm eines typischen Computersystems ist, das zum Implementieren einer Ausführungsform der vorliegenden Erfindung geeignet ist.

DETAILLIERTE BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN

[0022] Es wird nun detaillierter Bezug auf eine bevorzugte Ausführungsform der Erfindung genommen. In den begleitenden Zeichnungen wird ein Beispiel der bevorzugten Ausführungsform veranschaulicht. Während die Erfindung in Verbindung mit einer bevorzugten Ausführungsform beschrieben wird, wird verstanden, dass nicht gedacht ist, die Erfindung auf eine bevorzugte Ausführungsform zu begrenzen. Im Gegensatz dazu ist beabsichtigt, Alternativen, Modifikationen und Entsprechungen abzudecken, wie sie innerhalb des Geistes und Bereichs der Erfindung inkludiert sein können, wie durch die angefügten Ansprüche definiert.

[0023] Es werden ein Verfahren und ein System zum Managen von Softwareanwendungen und Diensten von einem zentralen Standort in einem Computernetz in den verschiedenen Zeichnungen beschrieben. In einem Computernetz mit großem Ausmaß mit vielen Servern und einer großen Endbenutzerbasis ist Management von Anwendungen und Software in dem Netz eine zeitraubende und fehleranfällige Aufgabe. Typischerweise installiert ein Systemadministrator eine neue Anwendung oder Dienst auf einem Diensthost, d.h. einem der Netzserver, was normalerweise auf dem Server geschieht. Information bezüglich Management der Anwendung, insbesondere der Standort und Namen von Dateien von Managementmodulen, wird manuell durch den Systemadministrator vermerkt. Diese Information wird dann in einem Administratorserver durch eine Administratorkonsole eingegeben. Sobald der Standort des neuen Anwendungsmanagementmoduls dem Administratorserver, z.B. einem Webserver, bekannt ist, können Endbenutzer auf die neue Anwendung zugreifen. Dieser Prozess wird umständlich und ineffizient, wenn es viele Server in dem Netz gibt, von denen jeder Anwendungen hat, die häufige Aktualisierung, Modifikation oder Austausch erfordern. Dieses Problem ist insbesondere aus der Sicht des Endbenutzers dadurch akut, dass die Erwartung hoch ist, dass eine Anwendung für eine Verwendung schnell verfügbar ist, nachdem sie empfangen wurde. Der beschriebene nicht-automatisierte zweistufige Prozess erhöht die Zeit, bevor eine Anwendung für Benutzer in dem Netz verfügbar sein kann.

[0024] Die vorliegende Erfindung ist ein Verfahren zum Automatisieren des Prozesses zum Registrieren neuer Anwendungen und Dienste in einem zentralen Managementstandort, wie etwa einem Webserver, wobei dadurch die Menge von Informationen, die sich ein Administrator merken muss, reduziert wird und ein Dienst Endbenutzern schneller zur Verfügung gestellt wird. In der beschriebenen Ausführungsform involviert die vorliegende Erfindung ein Managementkonsolenprogramm, das sich auf einem Administrati-

onsserver befindet, der andere Server oder Diensthosts in dem Netz managt, auch als Managementclients in dem Sinn bezeichnet, dass diese Server "Clients" des Administrationsservers sind. Die beschriebene Ausführungsform inkludiert auch einen persistenten Speicherbereich, der eine Datenbank zum Speichern von Managementinformation enthält, und (z.B. System- und Netzadministratoren) Authentifizierungsinformation bezüglich der Dienste auf den Diensthosts und ein "gut bekanntes" Verzeichnis, das mit jedem Managementclient in Verbindung steht, verwendet. In anderen bevorzugten Ausführungsformen, die nachstehend detaillierter beschrieben werden, können die Speicherbereiche z.B. über das Netz verteilt sein, anstatt von nur mit einem Server in Verbindung zu stehen. In einer anderen bevorzugten Ausführungsform befindet sich das Managementkonsolenprogramm nicht vollständig auf dem Administrationsserver, sondern kann auch zwischen dem Server und einer Administratorclientmaschine verteilt sein. Diese Komponenten werden in Fig. 2 gezeigt.

[0025] Fig. 2 ist ein Blockdiagramm von serverseitigen Komponenten eines Computernetzes in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung. Eine serverseitige Konfiguration **200** eines vollständigen Netzes (nicht gezeigt) kann betrachtet werden, zwei Sektionen, eine Sektion **202**, die eine Administrationsseite darstellt, und eine Sektion **204**, die Netzserver oder Serverhosts darstellt, aufzuweisen. Nicht gezeigt in Fig. 2 werden die Netzendbenutzer auf Clientmaschinen, die typischerweise auf Netzserver **206** zugreifen können, um Dienste vorzusehen, oder zum Ablaufen von Anwendungen oder zum Durchführen anderer Netzoperationen. Obwohl die Endbenutzer eines Computernetzes zu den Nutznießern der vorliegenden Erfindung dadurch gehören, dass Dienste und Anwendungen in dem Netz für sie schneller zur Verfügung stehen und nicht so häufig gestoppt werden, wird in der beschriebenen Ausführungsform die Erfindung durch einen Systemadministrator oder Netzmanager (d.h. den Benutzer) verwendet.

[0026] In der beschriebenen Ausführungsform werden Managementclients **206** durch einen Webserver **208** gemanagt. In anderen bevorzugten Ausführungsformen kann Server **208** ein anderer Typ eines Servers sein, wie etwa ein generischerer Administrationsserver, oder ein Server sein, der andere Funktionen hat, abhängig von der Größe des Netzes und der Kapazität des Servers. In jedem Fall hat Server **208** in dem Netz die Rolle zum Managen von Managementclients **206**. Ein Merkmal von Server **208** besteht darin, dass er ein Managementkonsolenprogramm **210** enthält, das nachstehend detaillierter beschrieben wird. Ein anderes Merkmal von Webserver **208** besteht darin, dass er Zugriff auf eine persistente Speicherbereichsdatenbank **212** hat, die Dienstmanagement-Modulinformation speichert. Webserver **208** kommuniziert mit Speicher **212** durch das Leichtgewichts-Verzeichniszugriffsprotokoll (LDAP, light-weight directory access protocol) **214**. In anderen bevorzugten Ausführungsformen können andere Datenzugriffsprotokolle zwischen Server **208** und Speicherbereich **212** verwendet werden. Speicherbereich **212** ist auch für Managementclients **206** zugreifbar. Der persistente Speicher **212** ist eine zuverlässige Datenbank, die Daten, in der beschriebenen Ausführungsform, in einem hierarchischen Format speichert. In anderen bevorzugten Ausführungsformen kann die Datenbank in einem relationalen Datenbankformat vorliegen oder Daten in einem Datendepot eines objektorientierten Typs speichern. Außerdem kann Speicher **212** in anderen bevorzugten Ausführungsformen über einen persistenten Speicherbereichsteil von Managementclients **206**, Webserver **208** und anderen persistenten Speichermedien, die in dem Netz verfügbar sind und auf die durch die Server zugegriffen werden kann, verteilt sein.

[0027] Wie erwähnt, wird die vorliegende Erfindung hauptsächlich durch einen Systemadministrator verwendet. Der Administrator greift auf Server **208** durch eine spezielle Client-Administratorkonsole **216** zu. In der beschriebenen Ausführungsform ist Konsole **216** mit einem webbasierten Browserprogramm ausgerüstet, das dem Administrator erlaubt, auf Server **208** zuzugreifen, und genauer Managementkonsolenprogramm **210** und Speicherbereich **212** zu verwenden. Server **208** kann auch als ein Managementkonsolenhost aus der Sicht von Browser-Host **216** bezeichnet werden. Wie nachstehend detaillierter beschrieben wird, kann ein Systemadministrator Browser-Host **216** verwenden, um Softwareanwendungen und Dienste auf Managementclients **206** zu managen.

[0028] Managementclients **206** können alle oder einige der Server in dem Netz inkludieren. Diese werden durch einen Systemadministrator durch Webserver **208** gemanagt, der mit Speicher **212** über LDAP kommuniziert. Jeder Managementclient hat einen oder mehr Dienste, die bei **218** gezeigt werden, und ein oder mehr entsprechende Managementmodule, die bei **220** im Serverhost **207** gezeigt werden. Wenn ein neuer Dienst installiert wird oder ein existierender Dienst aufgerüstet wird, wird ein Eintrag im Managementmodulbereich **220** geändert. Wie nachstehend detaillierter beschrieben wird, spiegelt sich diese Änderung in entsprechenden Einträgen im persistenten Speicher **212** wider. Obwohl Dienste **218** in Fig. 2 getrennt von Managementmodulen **220** gezeigt werden, sind die zwei Komponenten zueinander ganzheitlich. Mit anderen Worten ist ein Managementmodul eines Dienstes ganzheitlich mit dem Hauptkörper oder funktionalen Modulen des Dienstes gebunden. Die zwei Komponenten haben jedoch dennoch getrennte Rollen. Managementmodule **220** sind in Konfigurationsdateien gespeichert, ein Konfigurations-

komponentenverzeichnis wird nachstehend detaillierter beschrieben. In anderen bevorzugten Ausführungsformen kann die Information in Managementmodulen **220** in anderen Formaten gespeichert sein, wie etwa einer Datenbank oder einem Standardverzeichnis, dass auch andere Nicht-Management-Dateien enthält.

[0029] Die verbleibenden Komponenten in Fig. 2, die sich auf das Managementkonsolenprogramm beziehen, sprechen Authentifizierung und Zugriffssteuermerkmale an. Managementkonsolenprogramm **210** hat eine Authentifizierungsschicht **222**, die Benutzerverifizierung und Autorisierungsfunktionen durchführt, wie in Hinsicht auf Fig. 7 bis Fig. 9 nachstehend detaillierter beschrieben wird. Mit Konsolenhost **208** steht eine gemeinsame Gateway-Schnittstelle (Common Gateway Interface), oder CGI-Programm, in Verbindung, was durch einen Webserver verwendet wird, um Programme auszuführen. In der beschriebenen Ausführungsform wird CGI-Programm **224** verwendet, um Programme von Konsolenhost **208** auszuführen, und ist logisch in zwei Teile unterteilt: eine Managementkonsolen-CGI **226** und eine Servlet-CGI **228**. Managementkonsolen-CGI **226** kommuniziert mit Managementkonsolenprogramm **208** und wird nachstehend mit Bezug auf Fig. 8a und Fig. 8b detaillierter erläutert. Servlet-CGI **228** kommuniziert Authentifizierungsdaten von Konsolenhost **208** zu den Diensthosts **206**, und ist eine Komponente, die in der Technik gut bekannt ist.

[0030] Auf Diensthosts **206** befindet sich eine entsprechende Authentifizierungs- und Zugriffssteuerschicht **230**, die Teil von Managementmodulkomponente **220** ist. Authentifizierungsschicht **230** empfängt Daten von Konsolenhost **208** durch Servlet-CGI **228**. Diese Komponenten werden verwendet um sicherzustellen, dass ein Systemadministrator, der sich anmeldet, um das Managementkonsolenprogramm zu verwenden, um bestimmte Dienste zu managen, autorisiert ist, diese Dienste zu managen, und erlaubt auch einem "Super"-Systemadministrator, Administratoren und bestimmte Privilegien in dem Managementkonsolenrahmenwerk hinzuzufügen und zu löschen. In der beschriebenen Ausführungsform wird diese Funktionalität durch eine grafische Benutzerschnittstelle veranschaulicht, die in Fig. 7 gezeigt wird. Diensthosts **206** authentifizieren eine Zugriffssteuerung eines Benutzers und Autorisierung mit dem persistenten Datenspeicher **212** neu.

[0031] Fig. 3 ist ein Flussdiagramm, das einen Überblick über einen Prozess zum Registrieren eines neuen Dienstes in einem Netz in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung zeigt. Das Flussdiagramm zeigt die Schritte, die durch einen Systemadministrator unternommen werden, wenn entweder ein neuer Dienst registriert, ein Dienst aufgerüstet oder ein neuer Managementclient zu dem Netz hinzugefügt wird. In Schritt **302** wird ein Dienst auf einem bestimmten Managementclient installiert. Dies geschieht typischerweise durch eine Clientmaschine, die als ein Browserhost funktioniert, und wird gewöhnlich durch einen Systemadministrator durchgeführt. Ein Managementmodul, das mit dem Dienst in Verbindung steht, ist ein Segment von ausführbarem Code, das auch auf dem Managementclient installiert ist. Ein Beispiel eines Managementmoduls auf einem Mail-Server ist ein Modul, das eine maximale Quote (Kontingent) pro Endbenutzer anzeigt; d.h. die maximale Speichergröße, die ein Benutzer belegen darf. Ein anderes Beispiel ist ein Webserver, der sich im Besitz durch einen ISP (Internetdienstanbieter, Internet service provider) befindet, der Web-Standorte für seine Kunden unterbringt. In diesem Kontext kann ein Managementmodul die Hinzufügung eines neuen Web-Standorts auf dem Webserver managen.

[0032] Das Managementmodul kann eines von mehreren Typen sein. In der beschriebenen Ausführungsform sind die Typen von Managementmodulen browser-basiert, X-basiert und Kommandozeile. Ein browser-basiertes Managementmodul steht mit einer Anwendung in Verbindung, die in einem Web-Browser ausgeführt wird. Es wird vorausgesetzt, dass eine große Mehrheit der Anwendungstypen Anwendungen sein werden, die in einem Web-Browser laufen. Ein X-basiertes Managementmodul steht typischerweise mit einer autonomen Anwendung in Verbindung, die basierend auf dem X-Protokoll, einer Komponente des UNIX-Betriebssystems, läuft. Diese Anwendungen laufen im allgemeinen nicht innerhalb eines Browsers, sondern von einer Betriebssystem-Shell. Es ist aus Standard und gut bekannten X-Windows, einer UNIX-basierten grafische Benutzerschnittstelle, abgeleitet. Ein Kommandozeilen-Managementmodul steht mit einer Anwendung in Verbindung, die unter Verwendung von Kommandozeilen gemanagt wird, kann aber eingebettet sein in und ausgeführt werden von einem Web-Browser. Eine Kommandozeile kann oder kann nicht Laufzeitparameter aufweisen, wie nachstehend beschrieben wird. Beispiele von Kommandozeilenbefehlen sind "ls" (eine Liste von Dateien erhalten), "whoami" (Information über einen gegenwärtigen Benutzer zurückgeben) und "ps" (Information über einen Leistungsverhaltensstatus vorsehen). In anderen bevorzugten Ausführungsformen können andere Typen von Managementmodulen installiert sein.

[0033] In Schritt **304** registriert der Systemadministrator den Dienst und Managementmodule auf dem Managementclient. In der beschriebenen Ausführungsform geschieht dies durch Ablaufen eines Befehls, der als mc_reg bezeichnet wird, auf dem Managementclient. Durch Registrieren des Dienstes und der Managementmodule wird der Administrationsserver (Server **208** in Fig. 2) darüber informiert,

welcher Typ eines Moduls installiert wird. Typischerweise registriert ein Systemadministrator mehrere neue Dienste auf verschiedenen Managementclients. Somit werden Schritte **302** und **304** für mehrere Dienste auf verschiedenen Managementclients wiederholt. Sobald ein Dienst auf einem Serverhost registriert ist, werden bestimmte Dateien, die als Komponentenkonfigurationsdateien bezeichnet werden, die Managementdaten speichern, in einem Komponentenkonfigurationsverzeichnis auf dem Diensthost erstellt und gespeichert. Schritt **304** wird mit Bezug auf Fig. 4 detaillierter beschrieben.

[0034] In Schritt **306** wird eine Routine zum "Aufdecken" durch eine zugehörige grafische Benutzerschnittstelle initiiert, die mit Managementkonsolenprogramm **210** in Verbindung steht, und läuft auf einem Diensthost. Die Routine erlaubt dem Managementkonsolenprogramm, einen bestimmten Diensthost zu registrieren. Der Systemadministrator instruiert z.B. durch Browserhost **216** die Managementkonsole, zu einem bestimmten Diensthost oder einer Gruppe von Diensthosts zu gehen um nachzusehen, was registriert wurde. In der beschriebenen Ausführungsform geschieht dies durch die Managementkonsole durch Überprüfen eines gut bekannten Verzeichnisses, das als das Komponentenkonfigurationsverzeichnisses bezeichnet wird, auf den Diensthosts, die durch den Systemadministrator angezeigt werden. Schritt **306** wird in Fig. 5 detaillierter beschrieben. In einer bevorzugten Ausführungsform kann die Aufdeckungsroutine lokal auf dem Diensthost zu der Zeit laufen, zu der der Dienst in Schritt **302** installiert wird. Der Diensthost kann dann die Ergebnisse der entfernten oder Auto-Aufdeckung zu dem Managementkonsolenprogramm übertragen. In der beschriebenen Ausführungsform kann der Systemadministrator der Managementkonsole mitteilen, alle Diensthosts, die durch den Administrator kürzlich modifiziert, aufgerüstet oder neu hinzugefügt wurden, zu registrieren. In der beschriebenen Ausführungsform fährt das Managementkonsolenprogramm fort, diese Diensthosts zu überprüfen, und wird beliebige Aktualisierungen durch Überprüfung des Komponentenkonfigurationsverzeichnisses registrieren. Sobald alle modifizierten Diensthosts registriert wurden, können Endbenutzer beginnen, die Dienste oder Anwendungen zu verwenden, und der Registrationsprozess ist abgeschlossen.

[0035] Fig. 4 ist ein Flussdiagramm, das Schritt **304** von Fig. 3 zum Registrieren eines Dienstes in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung detaillierter zeigt. Schritt **304** hat den Prozess zum Registrieren eines neuen Dienstes auf einem Diensthost eingeführt, sodass die Managementkonsole später aufdecken kann, dass ein neuer Dienst auf diesem Host registriert wurde, wie durch einen Systemadministrator angewiesen. In Schritt **402** wird der Dienst- oder Anwendungstyp zu dem Diensthost identifiziert. Wie oben beschrieben, kann in der beschriebenen Ausführungsform ein Dienst einer von drei Typen sein: browser-basiert, X-basiert und Kommandozeile. In anderen bevorzugten Ausführungsformen können zusätzliche Typen eingegeben werden. In der beschriebenen Ausführungsform wird dieser Schritt auf dem Diensthost durchgeführt und ist ein Weg zum Informieren der Managementkonsole über den Anwendungstyp. In anderen bevorzugten Ausführungsformen kann diese Information in dem Browserhost eingegeben werden. Information, die auf dem Diensthost nach Schritt **402** eingegeben wird, hängt von dem Typ eines identifizierten Dienstes ab. Falls der Dienst Web-basiert ist, fährt das Flussdiagramm mit Schritt **404** fort. In Schritt **404** gibt der Systemadministrator den Standort des Managementmoduls des Dienstes auf dem Diensthost ein. In dem Fall von Web-basierten Diensten ist der Standort typischerweise in der Form eines einheitlichen Ressourcenlokators, oder URL. In Schritt **406** werden der Diensttyp und der URL des Managementmoduls als Parameter in einem gut bekannten Standort auf dem Diensthost gesichert. In der beschriebenen Ausführungsform werden diese zwei Elemente von Information, die als Komponenten bezeichnet werden, in einer UNIX-Datei, die als eine Komponentenkonfigurationsdatei bezeichnet wird, in dem Verzeichnis gesichert, das als Komponentenkonfigurationsverzeichnis bezeichnet wird. In anderen bevorzugten Ausführungsformen können andere Verzeichnisse auf dem Diensthost verwendet werden, um diese Komponenten zu speichern.

[0036] In Schritt **408** werden den zwei Komponenten, die in einem Dienstmanagementmodul enthalten sind, Komponentenidentifikatoren zugewiesenen. In der beschriebenen Ausführungsform besteht dies aus zwei Teilen: (1) einem eindeutigen Identifikator (wie etwa ein Solaris-Paketname, z.B. SUNWFTP), und (2) einer Versionsnummer. Somit werden dem URL und den Diensttypkomponenten ein Komponentenidentifikator zugewiesen und in einer Datei in dem Komponentenkonfigurationsverzeichnis gesichert. Außerdem wird ein "benutzerfreundlicher" Name für den Dienst, der bis zu diesem Punkt ein eindeutiger, aber langer und kryptischer Name war, eingegeben. Dieser benutzerfreundliche Name ist der Name, der auf der grafischen Benutzerschnittstelle angezeigt wird, wie nachstehend mit Bezug auf **Fig.** 6 detaillierter beschrieben wird. In Schritt **420** werden die Daten oder Komponenten, die in Schritten **406** und **408** beschrieben werden, in einer geeigneten Datei in dem Komponentenkonfigurationsverzeichnis gespeichert. Somit ist nach Schritt **420** alle Information, die benötigt wird, um Schritt **306** von Fig. 3 (den "Aufdeckungs"-Prozess) für einen Dienst eines Web-basierten Typs durchzuführen, in einer geeigneten Datei in einem gut bekannten Verzeichnis gespeichert, und der Prozess ist abgeschlossen.

[0037] Zurückkehrend zu Schritt **402** fährt, falls der Diensttyp X-basiert ist, die Steuerung mit Schritt **410** fort. Wie oben beschrieben, steht ein Dienst eines X-basierten Typs typischerweise mit einer autonomen Anwendung in Verbindung, die basierend auf dem X-Protokoll läuft, einer Komponente des UNIX-Betriebssystems. In Schritt **410** gibt der Systemadministrator den Pfad ein, der notwendig ist, um die X-basierte Anwendung aufzurufen. In Schritt **412** werden ein UNIX-Benutzer und eine Benutzergruppe eingegeben, um die X-basierte Anwendung aufzurufen. Die Steuerung geht dann zu Schritt **408**, wo dem Pfad, Benutzername und Gruppe Komponentenidentifikatoren zugewiesen werden. In Schritt **420** werden die Komponentenidentifikatoren in einer geeigneten Datei in dem Komponentenkonfigurationsverzeichnis gespeichert.

[0038] Für Managementmodule eines Kommandozeilentyps gibt der Systemadministrator ähnlich zu dem X-basierten Typ ein: einen Pfad, um die Kommandozeile aufzurufen, und einen UNIX-Benutzer und einen Gruppennamen, die notwendig sind, um die UNIX-Anwendung aufzurufen, wie in Schritt **414** gezeigt wird. In Schritt **416** bestimmt der Systemadministrator, ob es irgendwelche Laufzeitparameter in dem Befehl gibt (widergespiegelt in dem Managementmodul vom Kommandozeilentyp). Diese Parameter werden nicht zu der Zeit eingegeben, zu der der Dienst registriert wird, sondern zu der Zeit, zu der der Befehl durch den Endbenutzer ausgeführt oder laufen gelassen wird. Die grafische Benutzeroberfläche ist modifiziert oder angepasst widerzuspiegeln, ob der Endbenutzer Laufzeitparameter eingeben kann (z.B. Optionen, die der Benutzer zu der Zeit auswählen kann, zu der der Dienst verwendet wird). Falls es Laufzeitparameter gibt, stellt sie der Systemadministrator als Reaktion auf eine Aufforderung von der grafischen Benutzeroberfläche der Managementkonsole bereit. In Schritt **418** gibt der Systemadministrator statische Parameter ein, die durch den Befehl angefordert werden. Ein Managementmodul eines Kommandozeilentyps wird stets statische Parameter aufweisen, ungeachtet dessen, ob der Befehl Laufzeitparameter hat. Die Steuerung geht dann zu Schritt **408**, wo allen Daten Komponentenidentifikatoren zugewiesen werden, wie es für X-basierte und Web-basierte Managementmodule geschehen ist. Die Komponentenidentifikatoren werden dann in Dateien gesichert, die in Schritt **420** in dem Konfigurationskomponentenverzeichnis gespeichert werden. In der beschriebenen Ausführungsform hat der Dateiname das Format "Komponentenidentifikator – Versionsnummer", was eine Bestimmung der Nummer von Komponenten erleichtert, die in dem Verzeichnis registriert sind, wo jede Komponente eine Datei hat. In anderen bevorzugten Ausführungsformen kann der Dateiname in anderen Formaten auftreten, wo es eine Datei pro Befehl gibt, z.B. Komponentenidentifikator – Befehl #.

[0039] Fig. 5 ist ein Flussdiagramm, das Schritt **306** von Fig. 3 in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung detaillierter zeigt. In der beschriebenen Ausführungsform hat ein Diensthost ein laufendes Komponentensoftwaresegment, das alle Managementmodule der Dienste auf diesem Diensthost enthält. Das Komponentenkonfigurationsverzeichnis befindet sich in diesem Segment. Der Diensthost hat auch ein Managementkonsolen-Rahmenwerksegment, das Code enthält, der auch in dem Managementkonsolenprogramm enthalten ist, das sich auf dem Administrationsserver befindet. Z.B. befinden sich der Befehl mc_reg und ISP entfernter Shell-Code (ISP remote shell code), ein Programm zum entfernten Ausführen von X-basierten und Kommandozeilen-Managementprogrammen, auf sowohl der Managementkonsole als auch dem Diensthost. Fig. 5 beschreibt einen Aufdeckungsprozess, der das Komponentensoftwaresegment auf einem Diensthost nach Managementmodulen, die noch nicht registriert wurden, unter Verwendung von Software in dem Managementkonsolen-Rahmenwerksegment durchsucht.

[0040] In Schritt **502** spezifiziert ein Systemadministrator einen Diensthostnamen oder einen Dienstnamen durch eine grafische Benutzerschnittstelle auf dem Browserhost. Beispiele von grafischen Benutzerschnittstellen, die in der beschriebenen Ausführungsform verwendet werden, werden in **Fig.** 6 und Fig. 7 detaillierter gezeigt. Wie oben beschrieben, kann es viele Diensthosts geben, von denen jeder verschiedene verfügbare Dienste hat. Diese Wahlmöglichkeiten werden einem Systemadministrator durch eine Benutzerschnittstelle präsentiert. Typischerweise wird ein Administrator alle Diensthosts auswählen, die Dienste enthalten, die kürzlich modifiziert oder hinzugefügt wurden, und wird alle diese Diensthosts auf einmal von dem Browserhost eingeben. In Schritt **504** verbindet sich der Managementkonsolenhost mit dem einen oder mehr Diensthosts, die in Schritt **502** spezifiziert werden, um ein gut bekanntes Verzeichnis auf Komponentenkonfigurationsdateien abzutasten. In der beschriebenen Ausführungsform ist das gut bekannte Verzeichnis das Komponentenkonfigurationsverzeichnis. Die Managementkonsole kommuniziert mit dem Diensthost durch ein Standard-CGI- (gemeinsame Gateway-Schnittstelle) Programm, das typischerweise verwendet wird, um ein Web-basiertes Programm von einem Webserver zu initiieren, und ist in der Technik gut bekannt. In anderen bevorzugten Ausführungsformen kann das CGI-Programm nicht benötigt werden, falls der Administrationsserver nicht ein Web-basierte Server ist. Die Abtastung wird unter Verwendung eines Kommandozeilenprogramms durchgeführt, das Befehle über eine Netzverbindung sendet und sie auf dem Zielserver ausführen lässt. Genauer werden in der beschriebenen Ausführungsform die Befehle durch die Managementkonsole über

die Netzverbindung auf dem Diensthost ausgeführt. In der beschriebenen Ausführungsform geschieht dies durch ein ISP entferntes Shell-Protokoll. Somit wird während der Abtastung der UNIX-Befehl "Dateien auflisten", ls, in dem Komponentenkonfigurationsverzeichnis ausgeführt, um eine Liste der Komponentenkonfigurationsdateien zu erhalten. Eine Liste von Dateien, die mit der Managementkonsole registriert werden müssen, wird zu dem Administrationsserver gesendet.

[0041] In Schritt **506** untersucht die Managementkonsole die Liste von Dateien, die auf allen Diensthosts "aufgedeckt" wurden, die in Schritt **502** angegeben wurden. Es wird dann die gleiche Verbindung zwischen der Managementkonsole und den Diensthosts verwendet, um den Inhalt dieser Dateien abzufragen. In der beschriebenen Ausführungsform wird der UNIX-Befehl "concatenate", cat, auf dem Diensthost verwendet, um den Inhalt jeder Datei abzufragen. In anderen bevorzugten Ausführungsformen können ähnliche Befehle zum Abfragen des Inhalts einer Datei in anderen Betriebssystemen verwendet werden. Sobald der Inhalt von jeder Datei, die zu registrieren ist, von den Diensthosts abgefragt wurde, wird der Inhalt jeder einzelnen Datei unter Verwendung von Standard- und gut bekannten Parsing-Techniken durch die Managementkonsole auf dem Administrationsserver geparst. In der beschriebenen Ausführungsform ist eine Komponentenkonfigurationsdatei eine flache ASCII-Datei. Durch Parsen des Inhalts einer Datei werden der benutzerfreundliche Namen der Datei, Komponentenidentifikatoren und andere Befehlsausführungsinformation für jede Datei identifiziert. In der beschriebenen Ausführungsform spiegelt diese Information die Information wider, die in dem Komponentenkonfigurationsverzeichnis für jeden der drei Managementmodultypen gesichert wurde, wie in Fig. 4 gezeigt.

[0042] In Schritt **508** werden die Daten, die aus den Komponentenkonfigurationsdateien geparst wurden, in einem persistenten Speicherbereich gespeichert. Wie oben beschrieben, enthält eine Komponentenkonfigurationsdatei alle Information, die benötigt wird, um einen entsprechenden Dienst zu starten. Diese Information wird nun in einer Datenbank im persistenten Speicher gespeichert, auf den das Managementkonsolenprogramm und die Diensthosts zugreifen können. Ein Systemadministrator kann nun einen Dienst durch die Managementkonsole durch Modifizieren des Inhalts dieser Managementdaten des Dienstes, die in der persistenten und zuverlässigen Datenbank gespeichert sind, managen. In der beschriebenen Ausführungsform verbleiben Daten in dem persistenten Speicher, wenn das Netz heruntergefahren ist oder wenn die Managementkonsole nicht aktiv ist, und sind durch das leichtgewichtige Verzeichniszugriffsprotokoll (LDAP) zugreifbar. In anderen bevorzugten Ausführungsformen können alternative Zugriffsprotokolle abhängig von dem Typ eines verwendeten Speichers und des Netzes verwendet werden.

[0043] Fig. 6a bis 6c sind Bildschirmausdrucke einer grafischen Benutzerschnittstelle, die auf dem Browserhost angezeigt wird, in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung. Fig. 6a ist ein anfänglicher Bildschirmausdruck der Benutzerschnittstelle "Dienste registrieren". Ein Fenster **602** enthält ein Teilfenster **604** für einen Texteintrag, in dem ein Systemadministrator den Namen eines Diensthosts einträgt, auf dem sich Dienste befinden, die der Administrator zu administrieren wünscht. In der beschriebenen Ausführungsform gibt es einen Bereich, um einen Diensthost einzugeben. In anderen bevorzugten Ausführungsformen kann ein Administrator mehr als einen Diensthost eingeben. Auch wird ein Teilfenster für einen Texteintrag **606** gezeigt, in dem ein Administrator einen Diensthostnamen eingeben kann, der Dienste enthält, die der Administrator zu deregistrieren wünscht. Sobald die Auswahlen eingegeben wurden, kann der Benutzer auf einen Knopf **608** klicken, um eine Liste von Diensten abzufragen, für die der Benutzer autorisiert ist, sie auf diesem Diensthost zu managen. Der Administrator kann auch Knopf **610** drücken, um eine Liste von Diensten auf diesem Diensthost abzufragen, die deregistriert werden können.

[0044] Fig. 6b ist ein Bildschirmausdruck, der ein anderes Segment der Benutzerschnittstelle "Dienste registrieren" zeigt. Diese grafische Benutzerschnittstelle erlaubt einem Systemadministrator, Dienste auszuwählen, für die der Administrator autorisiert ist, sie zu managen. Benutzerautorisierung und Zugriffssteuerung werden nachstehend detaillierter beschrieben. Eine Liste von Diensten **612** wird in einem Fenster **614** angezeigt. Liste **612** ist aus Daten in Bezug auf den Benutzer abgeleitet, die in der Datenbank gespeichert sind, und enthält jene Dienste, die auf dem Diensthost verfügbar sind, der in Feld **604** von Fig. 6a eingetragen ist. Der Systemadministrator wählt jene Dienste aus, die er zu managen oder auf die er zuzugreifen wünscht. In der beschriebenen Ausführungsform wird dies mit einem Stern links von dem Dienstnamen gezeigt, wie etwa der Sun News (TM) Dienst **616**. Sobald der Dienst oder die Dienste ausgewählt wurden, klickt der Benutzer auf die Leiste "Oben ausgewählte Dienste registrieren" **618**. In der beschriebenen Ausführungsform geschieht dies unter Verwendung einer Zeigeeinrichtung, wie etwa einer Maus oder eines Trackballs, und ist in einer Fensterumgebung implementiert. In anderen bevorzugten Ausführungsformen kann eine nicht-grafische Benutzerschnittstelle, wie etwa eine einfache textbasierte Schnittstelle oder eine weiterentwickelte auf Spracherkennung basierte Schnittstelle, verwendet werden, um diese Information, ebenso wie die Information, die nachstehend mit Bezug auf die anderen Bild-

schirme beschrieben wird, einzugeben.

[0045] Wie oben beschrieben, inkludiert ein Managementkonsolenprogramm der vorliegenden Erfindung ein Verfahren einer "einzelnen Anmeldung" für Benutzerauthentifizierung und Zugriffssteuerung, die einen Nutzen aus einer zentralen Managementkonsole zum Managen von Diensten auf vielen Diensthosts in einem verteilten Web-basierten Netz ziehen. Gegenwärtig muss sich in Web-basierten Netzen ein Systemadministrator, der zum Unterhalten von Diensten verantwortlich ist, die auf vielen Diensthosts verfügbar sind, neu authentifizieren und die Beglaubigungen des Administrators zu jedem Diensthost weitergeben, an dem sich der Administrator anmeldet. Dies trifft zu, da der Administrator, der von einem Browser arbeitet, nicht eine einzelne universelle Identität hat, die für Authentifizierung verwendet werden kann. Hier verweist Authentifizierung auf verifizierende Beglaubigungen und Autorisierungen eines Benutzer, bevor ihm erlaubt wird, einen bestimmten Diensthost zu managen, oder genauer Operationen zum Managen von Diensten auf einem bestimmten Diensthost durchzuführen. Es ist notwendig, ein konsistentes Verständnis überall in dem Netz davon zu haben, wer der Benutzer ist und was dem Benutzer erlaubt ist, auf den Diensthosts zu tun.

[0046] Die vorliegende Erfindung erlaubt zentralisiertes Management und einzelne Anmeldung eines Benutzers für Authentifizierung bezüglich Management von Diensten auf Diensthosts von einem Browserhost. Das Managementkonsolenprogramm **210** von Fig. 2 enthält eine Autorisierungs- und Zugriffssteuerkomponente oder Schicht **222**. Diese Autorisierungsschicht greift auf Benutzerdaten aus der Datenbank **212** für eine Verifizierung zu und kommuniziert diese Information zu entsprechenden Autorisierungs- oder Authentifizierungsschichten **230** auf einem Diensthost **206**. Die Information wird behandelt und zu jedem Diensthost übertragen, den ein Systemadministrator zu managen wünscht, ohne dass sich der Administrator an jedem einzelnen Diensthost neu authentifizieren muss.

[0047] Information in Bezug auf jeden Benutzer wird in Datenbank **212** gespeichert, und Information, die durch einen Benutzer eingegeben wird, wird gegenüber dieser Information authentifiziert. Die Information, oder Beglaubigungen, wird, falls verifiziert, durch ein CGI-Programm zu den Diensthosts weitergegeben, die durch den Benutzer angezeigt werden. Sobald durch die Diensthosts empfangen, wird die Information erneut gegenüber dem Benutzerprofil in der Datenbank im Namen des Systemadministrators authentifiziert; mit anderen Worten geschieht dies "hinter der Bühne", ohne Eingriff oder beliebige zusätzliche Schritte von dem Benutzer. Der Benutzer muss sich nur bei der Managementkonsole durch einen Browser einmal anmelden (d.h. gewisse Information eingeben, wie etwa Name und Passwort), und diese Information wird automatisch zu den Diensthosts weitergegeben.

[0048] Fig. 7 ist ein Bildschirmausdruck einer grafischen Benutzerschnittstelle bezüglich der Zugriffssteuerung und Authentifizierung eines Benutzers des Managementkonsolenprogramms in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung. Ein Fenster **702** hat die Überschrift "Administratoren managen". Dieses Fenster wird verwendet, um neue Administratoren und zugehörige Passwörter und Dienste einzutragen, für die dem neuen Administrator erlaubt wird, sie zu managen. Innerhalb von Fenster **702** gibt es ein Teilfenster **704** zum Eingeben eines Administratornamens und Teilfenster **706** und **708** zum Eingeben und erneuten Eingeben eines Passworts. In dem unteren Abschnitt von Fenster **702** enthält ein anderes Teilfenster **710** eine Liste von Diensten, die dem Administrator, der in Teilfenster **704** eingetragen ist, erlaubt werden zu managen. Sobald die Dienste durch den verwaltenden oder "Super"-Administrator ausgewählt sind, wird der Knopf **712** gedrückt.

[0049] Fig. 8a und Fig. 8b sind Flussdiagramme eines Prozesses zum Durchsetzen von Zugriffssteuerung und Autorisierung in dem Managementsteuerprogramm in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung. Der Durchsetzungsprozess beginnt mit einem Benutzer, der den Browserhost (d.h. Administrationskonsole **216** von Fig. 2) auf einen URL des Managementkonsolenhosts richtet. Somit gibt der Benutzer in Schritt **802** den URL des Konsolenhosts von dem Browserhost ein. Der URL für die Managementkonsole ist in der Form eines Standard-URL in einem webbasierten Netz. In anderen bevorzugten Ausführungsformen können andere Typen von Lokatoren abhängig von dem Typ des Netzes verwendet werden.

[0050] In Schritt **804** wird von dem Administrator/Benutzer ein Benutzername und ein Passwort zum Zugriff auf das Managementkonsolenprogramm auf dem Konsolenhost angefordert. In Schritt **806** akzeptiert die Managementkonsole den Benutzernamen und das Passwort, die in Schritt **804** eingegeben werden, und der Benutzer ist authentifiziert. Dieser Schritt wird detaillierter in Fig. 9 beschrieben. Die Managementkonsole zeigt die Dienste auf einem ausgewählten Diensthost an, wie im Bereich **612** von **Fig.** 6 gezeigt, für die der Benutzer autorisiert ist, sie zu managen, durch Untersuchen von Daten in Datenbank **212**. Dies geschieht durch Verwenden des Managementkonsolensegmentes der CGI, wie in Fig. 2 gezeigt. In der beschriebenen Ausführungsform ist die Autorisierung eines Administrators im Sinne von Diensten definiert, für die dem Administrator erlaubt wird, sie zu managen. Während dieses Schrittes baut die Managementkonsole einen URL für jeden Dienst

und Host auf, für die dem Administrator erlaubt wird, ihn zu managen. Dieser Prozess wird auch detaillierter mit Bezug auf Fig. 9 beschrieben. Die URLs erlauben dem Konsolenhost, jeden Diensthost und Dienst zu lokalisieren, der durch den Administrator gemanagt werden kann.

[0051] In Schritt **808** wählt der Benutzer eine Instanz eines Dienstes (d.h. einen bestimmten Dienst von einem Diensthost), den der Benutzer wünscht zu managen. Ein Dienst kann sich auf mehreren unterschiedlichen Diensthosts befinden, sodass der Benutzer eine Instanz eines Dienstes von einem bestimmten Diensthost wählen muss. Durch Auswählen des benutzerfreundlichen Namens hat der Benutzer einen der URLs ausgewählt, die in Schritt **806** aufgebaut werden. In Schritt **810** initiiert der Managementkonsolenhost die Servlet-CGI-Komponente der CGI. In der beschriebenen Ausführungsform geschieht dies durch Vergleichen der Benutzerbeglaubigungen oder des Profils gegenüber der Authentifizierung des Benutzers und Zugriffssteuerdaten in der Datenbank. Diese Verifizierung wird durchgeführt, bevor eine Verbindung zu dem Diensthost durch Servlet-CGI **224** hergestellt wird, als eine zusätzliche Vorsichtsmaßnahme gegen Benutzer, die versuchen, Dienste auf diesem Diensthost zu managen, ohne durch Managementkonsolenhost **208** zu gehen. Da dies eine Netzumgebung ist, ist es für einen Benutzer möglich, die Konsolenhost-Verifizierungsschritte zu umgehen und zu versuchen, auf Dienste auf einem Diensthost direkt von einer Clientmaschine anstatt von Browserhost **216** von Fig. 2 zuzugreifen. So werden die Benutzerbeglaubigungen durch die Servlet-CGI gegen die Benutzerdaten verglichen, die in Datenbank **212** gespeichert sind.

[0052] In Schritt **812** verwendet die Servlet-CGI eine Standardprozedur zum Weitergeben der Benutzerbeglaubigungen zu dem Diensthost oder Hosts, die durch den Benutzer angezeigt werden. In der beschriebenen Ausführungsform führt der Diensthost, sobald die Daten empfangen sind, Authentifizierung und Zugriffssteuerung unter Verwendung der Daten durch ihren Vergleich gegen Daten in der Datenbank durch. In anderen bevorzugten Ausführungsformen kann dieser Schritt abhängig von unabhängigen Sicherheitsmerkmalen, die in dem bestimmten Netz verfügbar sind, das das Managementkonsolenprogramm implementiert, nicht notwendig sein. Diese erneute Authentifizierung geschieht ohne jeglichen Eingriff von dem Benutzer und wird durchgeführt um sicherzustellen, dass ein Benutzer nicht versucht, sich direkt an dem Diensthost anzumelden, wobei dadurch die Authentifizierungs- und Zugriffssteuerschicht des Managementkonsolenhosts umgangen wird. Somit kann durch Durchführen einer zweiten Prüfung gegen die Datenbank, ohne dass es erforderlich ist, dass der Benutzer beliebige zusätzliche Operationen durchführt, die Managementkonsole ein sicheres Management von Diensten in dem Netz sicherstellen. Falls die erneute Authentifizierung in Schritt **814** erfolgreich ist, erlaubt das Managementkonsolenprogramm auf dem Konsolenhost dem Benutzer, Managementoperationen in dem gewählten Dienst oder Diensten von dem Browser durchzuführen, wie in Schritt **816** gezeigt, in welchem Punkt der Durchsetzungsprozess abgeschlossen ist. Falls die erneute Authentifizierung nicht erfolgreich ist, wird dem Benutzer Autorität versagt, den ausgewählten Dienst zu managen, und ihm wird der Anmeldebildschirm erneut gezeigt.

[0053] Fig. 9 ist ein Flussdiagramm, das Schritt **806** von Fig. 8a detaillierter zeigt. In Schritt **806** wird der Benutzer authentifiziert und die Dienste, für die der Benutzer autorisiert ist, auf sie zuzugreifen, werden bestimmt und die URLs zu jedem dieser Dienste werden aufgebaut. In Schritt **902** authentifiziert der Managementkonsolenhost den Benutzer durch Abfragen von Information in Bezug auf den Benutzer aus der Datenbank. Diese Information besteht aus Name und Passwort des Benutzers. Sobald der Benutzername und das Passwort verifiziert sind, wird eine Liste von Diensten abgeleitet, für die der Benutzer autorisiert ist, sie zu managen. In Schritt **904** initiiert der Konsolenhost das Managementkonsolensegment **226** des CGI-Programms mit den Benutzerbeglaubigungen, die in Schritt **902** verifiziert wurden. Wie oben beschrieben, ist dies der erste Schritt beim Herstellen einer Verknüpfung mit einem Diensthost.

[0054] Die andere Komponente der CGI ist die Servlet-CGI (Element **224** von Fig. 2) und wird verwendet, um die Verbindung mit dem Diensthost herzustellen. In Schritt **906** fragt die Managementkonsolen-CGI Datenbank **212** von Fig. 2 ab, um die Liste von Diensten zu erhalten, für die der Benutzer autorisiert ist, sie zu managen. Verknüpfungen zu diesen Diensten werden in der Form von URLs zu allen Diensten auf der Liste aufgebaut. Die Datenbank enthält einen Eintrag für jeden Benutzer, der Information enthält, inkludierend Name, Passwort, Grad (z.B. Super-Systemadministrator) des Benutzers und eine Liste von Diensten, für die dem Benutzer erlaubt ist, sie zu managen. Ein Super-Systemadministrator kann alle Dienste managen und Zugriffssteuerparameter für die anderen Benutzer (z.B. Junior-Systemadministratoren) definieren. Die Liste von Diensten enthält "benutzerfreundliche" Namen der Dienste (auch in der Datenbank enthalten) an Stelle des Dienst-URL. Die Steuerung kehrt dann zu Schritt **806** von Fig. 8a zurück, wo der Benutzer aus der Liste von Diensten auswählt, welche Dienste er zu managen wünscht.

[0055] Die vorliegende Erfindung setzt verschiedene computerimplementierte Operationen ein, die Daten involvieren, die in Computersystemen gespeichert sind. Diese Operationen inkludieren, sind aber

nicht darauf begrenzt, jene, die physische Manipulation von physischen Quantitäten erfordern. Gewöhnlich, obwohl nicht notwendigerweise, nehmen diese Quantitäten die Form von elektrischen oder magnetischen Signalen an, die fähig sind, gespeichert, transferiert, kombiniert, verglichen oder anderweitig manipuliert zu werden. Die hierin beschriebenen Operationen, die einen Teil der Erfindung bilden, sind nützliche Maschinenoperationen. Die durchgeführten Manipulationen werden häufig auch in Begriffen bezeichnet, wie etwa Erzeugen, Identifizieren, Laufen, Bestimmen, Vergleichen, Ausführen, Herunterladen oder Erfassen. Es ist manchmal zweckmäßig, hauptsächlich aus Gründen einer gemeinsamen Verwendung, auf diese elektrischen oder magnetischen Signale als Bits, Werte, Elemente, Variablen, Zeichen, Daten oder dergleichen zu verweisen. Es sollte jedoch daran erinnert werden, dass alle diese und ähnliche Begriffe mit den geeigneten physischen Quantitäten zu verbinden sind und lediglich zweckmäßige Kennzeichnungen sind, die auf diese Quantitäten angewendet werden.

[0056] Die vorliegende Erfindung bezieht sich auch auf eine Einrichtung, ein System oder eine Vorrichtung, wie etwa einen Browserhost **216** und einen Managementkonsolenhost **208**, zum Durchführen der zuvor erwähnten Operationen. Das System kann speziell für die erforderlichen Zwecke aufgebaut sein, oder kann ein Mehrzweckcomputer sein, der durch ein Computerprogramm, das in dem Computer gespeichert ist, selektiv aktiviert oder konfiguriert wird. Die oben präsentierten Prozesse beziehen sich nicht inhärent auf irgendeinen bestimmten Computer oder eine andere Berechnungsvorrichtung. Insbesondere können verschiedene Mehrzweckcomputer mit Programmen verwendet werden, die in Übereinstimmung mit den Unterweisungen hierin geschrieben werden, oder es kann alternativ zweckmäßiger sein, ein spezialisierteres Computersystem aufzubauen, um die erforderlichen Operationen durchzuführen.

[0057] Fig. 10 ist ein Blockdiagramm eines Mehrzweckcomputersystems **1000**, das zum Ausführen der Verarbeitung in Übereinstimmung mit einer Ausführungsform der vorliegenden Erfindung geeignet ist. Das Managementkonsolenprogramm, inkludierend die Authentifizierungs- und Zugriffssteuerschicht, kann sich auf einem derartigen Mehrzweckcomputer befinden. Außerdem kann Browserhost **216** ein derartiger Mehrzweckcomputer sein. Fig. 10 veranschaulicht eine Ausführungsform eines Mehrzweckcomputersystems. Es können andere Computersystemarchitekturen und Konfigurationen zum Ausführen der Verarbeitung der vorliegenden Erfindung verwendet werden. Computersystem **1000**, das aus verschiedenen nachstehend beschriebenen Teilsystemen besteht, inkludiert mindestens ein Mikroprozessor-Teilsystem (auch als eine zentrale Verarbeitungseinheit oder CPU bezeichnet) **1002**. D.h. CPU **1002** kann durch einen Einchip-Prozessor oder durch mehrere Prozessoren implementiert werden. CPU **1002** ist ein Mehrzweck-Digitalprozessor, der den Betrieb des Computersystems **1000** steuert. Unter Verwendung von Instruktionen, die aus dem Speicher abgerufen werden, steuert die CPU **1002** den Empfang und die Manipulation von Eingabedaten, und die Ausgabe und Anzeige von Daten auf Ausgabeeinrichtungen.

[0058] CPU **1002** ist bidirektional mit einem ersten primären Speicher **1004**, typischerweise einem Speicher mit wahlfreiem Zugriff (RAM), und unidirektional mit einem zweiten primären Speicherbereich **1006**, typischerweise ein Nur-Lesespeicher (ROM), über einen Speicherbus **1008** gekoppelt. Wie in der Technik gut bekannt ist, kann der primäre Speicher **1004** als ein allgemeiner Speicherbereich und als ein Notizblockspeicher verwendet werden, und kann auch verwendet werden, um Eingabedaten und verarbeitete Daten zu speichern. Er kann auch Programmierinstruktionen und Daten speichern, z.B. in der Form einer hierarchischen Datenbank, wie etwa Datenbank **212**, zusätzlich zu anderen Daten und Instruktionen für Prozesse, die in CPU **1002** arbeiten, und wird typischerweise für eine schnelle Übertragung von Daten und Instruktionen auf eine bidirektionale Art und Weise über den Speicherbus **1008** verwendet. Wie in der Technik auch gut bekannt ist, inkludiert der primäre Speicher **1006** typischerweise Basisbetriebsinstruktionen, Programmcode, Daten und Objekte, die durch die CPU **1002** verwendet werden, um ihre Funktionen durchzuführen. Die primären Speichereinrichtungen **1004** und **1006** können beliebige geeignete computerlesbare Speichermedien inkludieren, die nachstehend beschrieben werden, abhängig z.B. davon, ob Datenzugriff bidirektional oder unidirektional sein muss. CPU **1002** kann auch direkt und sehr rasch häufig benötigte Daten abfragen und in einem Cache-Speicher **1010** speichern.

[0059] Eine entfernbare Massenspeichereinrichtung **1012** sieht zusätzliche Datenspeicherkapazität für das Computersystem **1000** vor, und ist entweder bidirektional oder unidirektional mit CPU **1002** über einen peripheren Bus **1014** gekoppelt. Z.B. gibt eine spezifische entfernbare Massenspeichereinrichtung, die gewöhnlich als eine CD-ROM bekannt ist, typischerweise Daten unidirektional zu der CPU **1002** weiter, wohingegen eine Diskette Daten bidirektional zu der CPU **1002** weitergeben kann. Speicher **1012** kann auch computerlesbare Medien inkludieren, wie etwa ein magnetisches Band, Flash-Speicher, Signale, die in einer Trägerwelle verkörpert sind, PC-CARDS, tragbare Massenspeichereinrichtungen, holografische Speichereinrichtungen und andere Speichereinrichtungen. Ein fester Massenspeicher **1016** sieht auch zusätzliche Datenspeicherkapazität vor und ist bidirektional mit CPU **1002** über den peripheren Bus **1014** gekoppelt. Das häufigste Beispiel

von Massenspeicher **1016** ist ein Festplattenlaufwerk. Allgemein ist Zugriff auf diese Medien langsamer als Zugriff auf die primären Speicher **1004** und **1006**. Massenspeicher **1012** und **1016** speichern allgemein zusätzliche Programmierinstruktionen, Daten und dergleichen, die typischerweise durch die CPU **1002** nicht aktiv verwendet werden. Es wird erkannt, dass die Information, die innerhalb von Massenspeicher **1012** und **1016** beibehalten wird, falls notwendig, auf eine Standardweise als Teil vom primären Speicher **1004** (z.B. RAM) als virtueller Speicher einbezogen werden kann.

[0060] Zusätzlich zum Versehen von CPU **1002** mit Zugriff auf Speicherteilsysteme wird der periphere Bus **1014** verwendet, um Zugriff auf andere Teilsysteme und ebenso Einrichtungen vorzusehen. In der beschriebenen Ausführungsform inkludieren diese einen Anzeigemonitor **1018** und Adapter **1020**, eine Druckereinrichtung **1022**, eine Netzschnittstelle **1024**, eine unterstützende Eingabe-/Ausgabe-Einrichtungsschnittstelle **1026**, eine Soundkarte **1028** und Lautsprecher **1030** und andere Teilsysteme, je nach Notwendigkeit.

[0061] Die Netzschnittstelle **1024** erlaubt CPU **1002**, mit einem anderen Computer, Computernetz oder Telekommunikationsnetz unter Verwendung einer Netzverbindung, wie gezeigt, gekoppelt zu werden. Durch die Netzschnittstelle **1024** wird betrachtet, dass die CPU **1002** Information, z.B. Datenobjekte oder Programminstruktionen, von einem anderen Netz empfangen kann, oder Information zu einem anderen Netz im Verlauf einer Durchführung der oben beschriebenen Verfahrensschritte ausgeben kann. Information, die häufig als eine Sequenz von Instruktionen dargestellt wird, die in einer CPU auszuführen sind, kann z.B. in der Form eines Computerdatensignals, das in einer Trägerwelle verkörpert ist, von/zu einem anderen Netz empfangen und ausgegeben werden. Eine Schnittstellenkarte oder eine ähnliche Einrichtung und geeignete Software, die durch CPU **1002** implementiert wird, können verwendet werden, um das Computersystem **1000** mit einem externen Netz zu verbinden und Daten gemäß Standardprotokollen zu übertragen. Das heißt Verfahrensausführungsformen der vorliegenden Erfindung können ausschließlich in CPU **1002** ausgeführt werden, oder können über ein Netz, wie etwa das Internet, Intra-Netze oder lokale Netze, in Verbindung mit einer entfernten CPU ausgeführt werden, die an einem Abschnitt der Verarbeitung teilhat. Zusätzliche Massenspeichereinrichtungen (nicht gezeigt) können auch mit einer CPU **1002** über Netzschnittstelle **1024** verbunden sein.

[0062] Hilfs-E/A-Einrichtungsschnittstelle **1026** repräsentiert allgemeine und angepasste Schnittstellen, die der CPU **1002** erlauben, Daten zu/von anderen Einrichtungen zu senden und typischer zu empfangen, wie etwa Mikrofone, berührungsempfindliche Anzeigen, Transducer-Kartenleser, Bandlesegeräte, Sprach- der Handschrifterkennungsgeräte, biometrische Leseeinrichtungen, Kameras, tragbare Massenspeichereinrichtungen und andere Computer.

[0063] Mit der CPU **1002** ist auch eine Tastatursteuervorrichtung **1032** über einen lokalen Bus **1034** zum Empfangen einer Eingabe von einer Tastatur **1036** oder einer Zeigereinrichtung **1038** und Senden dekodierter Symbole von der Tastatur **1036** oder Zeigereinrichtung **1038** zu der CPU **1002** gekoppelt. Die Zeigereinrichtung kann eine Maus, ein Stylus, ein Trackball oder ein Tablett sein, und ist zum Interagieren mit einer grafischen Benutzerschnittstelle von Nutzen.

[0064] Außerdem beziehen sich Ausführungsformen der vorliegenden Erfindung ferner auf Computerspeicherprodukte mit einem computerlesbaren Medium, die Programmcode zum Durchführen verschiedener auf einem Computer implementierter Operationen enthalten. Das computerlesbare Medium ist eine beliebige Datenspeichereinrichtung, die Daten speichern kann, die danach durch ein Computersystem gelesen werden können. Die Medien und der Programmcode können diejenigen sein, die speziell für die Zwecke der vorliegenden Erfindung gestaltet und aufgebaut sind, oder sie können von der Art sein, die einem gewöhnlichen Durchschnittsfachmann für Softwaretechnik gut bekannt sind. Beispiele von computerlesbaren Medien inkludieren, sind aber nicht darauf begrenzt, alle oben erwähnten Medien: magnetische Medien, wie etwa Festplatten, Disketten und magnetische Bänder; optischen Medien, wie etwa CD-ROM-Scheiben; magneto-optische Medien, wie etwa floptical Disks; und speziell konfigurierte Hardwareeinrichtungen, wie etwa anwendungsspezifische integrierte Schaltungen (ASICs), programmierbare Logikeinrichtungen (PLDs) und ROM- und RAM-Einrichtungen. Das computerlesbare Medium kann auch als ein Datensignal, das in einer Trägerwelle verkörpert ist, über ein Netz von gekoppelten Computersystemen verteilt werden, sodass der computerlesbare Code auf eine verteilte Weise gespeichert und ausgeführt wird. Beispiele von Programmcode inkludieren sowohl Maschinencode, wie z.B. durch einen Compiler erzeugt, als auch Dateien, die Code höherer Ebene enthalten, der unter Verwendung eines Interpreters ausgeführt werden kann.

[0065] Es wird durch einen Durchschnittsfachmann erkannt, dass die oben beschriebenen Hardware- und Softwareelemente von Standardgestaltung und Aufbau sind. Andere Computersysteme, die zur Verwendung mit der Erfindung geeignet sind, können zusätzliche oder weniger Teilsysteme inkludieren. Außerdem veranschaulichen der Speicherbus **1008**, der periphere Bus **1014** und der lokale Bus **1034** ein beliebiges Zusammenschaltungsschema, das dazu

dient, die Teilsysteme zu verknüpfen. Ein lokaler Bus könnte z.B. verwendet werden, um die CPU mit einem festen Massenspeicher **1016** und einem Anzeigeadapter **1020** zu verbinden. Das in Fig. 10 gezeigte Computersystem ist nur ein Beispiel eines Computersystems, das zur Verwendung mit der Erfindung geeignet ist. Es können auch andere Computerarchitekturen mit unterschiedlichen Konfigurationen von Teilsystemen genutzt werden.

[0066] Obwohl die vorangehende Erfindung einigermaßen detailliert für die Zwecke einer Klarheit des Verständnisses beschrieben wurde, wird offensichtlich, dass gewisse Änderungen und Modifikationen innerhalb des Bereichs der angefügten Ansprüche praktiziert werden können. Des weiteren sollte vermerkt werden, dass es alternative Wege zum Implementieren von sowohl dem Prozess als auch der Vorrichtung der vorliegenden Erfindung gibt. Obwohl die Erfindung beschrieben wurde, einen Webserver als den Administrationsserver zu verwenden, kann z.B. auch ein nicht-webbasierter Server verwendet werden, um das Managementkonsolenprogramm ablaufen zu lassen. In einem anderen Beispiel kann Datenbank **212** eine verteilte Datenbank sein, die auf dem Konsolenhost und verschiedenen Diensthosts an Stelle von in einer einzelnen persistenten Datenbank gespeichert wird. In noch einem anderen Beispiel können Datenabfrageprotokolle mit Ausnahme von LDAP verwendet werden, um Daten aus Datenbank **212** oder aus einer flachen Datei, die in einem persistenten Speicherbereich gespeichert ist, abzufragen. In noch einem anderen Beispiel kann die Aufdeckungsroutine "lokal" auf einem Diensthost laufen, während der Dienst installiert wird, an Stelle von zu einem späteren Zeitpunkt auf dem Konsolenhost. Entsprechend sind die vorliegenden Ausführungsformen als veranschaulichend und nicht beschränkend zu betrachten, und die Erfindung ist nicht auf die hierin angegebenen Details zu begrenzen, sondern kann innerhalb des Bereichs der angefügten Ansprüche modifiziert werden.

Patentansprüche

1. Verfahren zum Sichern von Zugriff auf die Administration einer Vielzahl von verschiedenen Diensten (**218**), die sich auf einem oder mehr Diensthostcomputern (**206**) befinden, von einem Administrationsservercomputer (**208**), der mit dem einem oder mehr Diensthostcomputern (**206**) verbunden ist, wobei es einen Dienstmanager (**208**) gibt, der sich auf dem Administrationsserver befindet, das Verfahren umfassend:
Vorsehen eines ausgewählten Benutzeridentifikators und eines entsprechenden privaten Schlüsselwortes, wobei der Benutzeridentifikator angeordnet ist, einen Benutzer zu identifizieren, der administrativen Zugriff auf mindestens einen der verschiedenen Dienste hat;
Authentifizieren des Benutzers durch Vergleichen des ausgewählten Benutzeridentifikators und des entsprechenden privaten Schlüsselwortes gegen eine Vielzahl von Benutzeridentifikatoren und privaten Schlüsselwörtern, die in einem persistenten Speicherbereich (**212**) gespeichert sind, wobei der Vergleich unter Steuerung des Dienstmanagers (**210**) durchgeführt wird;
Ableiten einer Liste von Diensten, auf die der Benutzer, der mit dem Benutzeridentifikator in Verbindung steht, administrativen Zugriff hat;
wenn eine Anfrage durchgeführt wird, einen ausgewählten der Dienste in der abgeleiteten Liste von Diensten zu administrieren, Verifizieren in dem Diensthostcomputer (**206**), der mit dem ausgewählten Dienst in Verbindung steht, dass dem Benutzer, der mit dem ausgewählten Benutzeridentifikator in Verbindung steht, gestattet ist, auf den ausgewählten Dienst zuzugreifen, durch Untersuchen von Zugriffssteuerdaten, die mit dem ausgewählten Benutzeridentifikator in dem persistenten Speicherbereich (**212**) in Verbindung stehen, und
Transferieren von einer oder mehr Managementdateien auf dem Diensthostcomputer (**206**) zu dem Administrationsserver (**208**), wobei dadurch eine Manipulation der Managementdateien unter Nutzung des Dienstmanagers (**210**) erleichtert wird.

2. Verfahren, wie in Anspruch 1 vorgetragen, worin der Administrationsservercomputer (**208**) mit einem Administrationsclientcomputer (**216**), der zum Ablaufen eines Browserprogramms geeignet ist, verbunden ist, und worin der ausgewählte Benutzeridentifikator und das entsprechende privaten Schlüsselwort über eine Kommunikationsverbindung zwischen dem Administrationsclientcomputer (**216**) und dem Administrationsservercomputer (**208**) vorgesehen sind, wobei die Kommunikationsverbindungen zwischen dem Administrationsservercomputer, dem Administrationsclientcomputer und dem einen oder mehr Diensthostcomputern ein Internetprotokoll nutzen.

3. Verfahren, wie in Anspruch 1 vorgetragen, wobei Vorsehen eines ausgewählten Benutzeridentifikators und eines entsprechenden privaten Schlüsselwortes ferner Anmelden bei dem Dienstmanager (**210**) durch den Administrationsclientcomputer (**216**) umfasst.

4. Verfahren, wie in Anspruch 1 vorgetragen, wobei Authentifizieren des Benutzers ferner Nutzen eines leichtgewichtigen Verzeichniszugriffsprotokolls umfasst, um den Benutzeridentifikator und das entsprechende private Schlüsselwort zu dem persistenten Speicherbereich (**212**) zu kommunizieren.

5. Verfahren, wie in Anspruch 1 vorgetragen, wobei jeder Benutzeridentifikator ein entsprechendes Benutzerprofil hat, das eine globale Benutzeridentität entsprechend einem bestimmten Dienstmanagerbe-

nutzer darstellt.

6. Verfahren, wie in Anspruch 1 vorgetragen, wobei Ableiten einer Liste von Diensten ferner Durchsuchen des persistenten Speicherbereichs (**212**) umfasst, wobei der persistente Speicherbereich (**212**) eine Benutzerprofildatenbank enthält, inkludierend für jeden Benutzer einen Benutzerzugriffsgrad, eine Liste von zulässigen Diensten und ein Passwort.

7. Verfahren, wie in Anspruch 1 vorgetragen, wobei Verifizieren in dem Diensthostcomputer (**206**), dass dem Benutzer, der mit dem ausgewählten Benutzeridentifikator in Verbindung steht, gestattet ist, auf den ausgewählten Dienst aus der Liste von Diensten zuzugreifen, ferner Kommunizieren des ausgewählten Benutzeridentifikators und des entsprechenden privaten Schlüsselwortes zu dem Hostservercomputer (**207**) unter Verwendung einer gemeinsamen Gateway-Schnittstelle (**226**) umfasst.

8. Verfahren, wie in Anspruch 1 vorgetragen, wobei der Diensthostcomputer (**206**) ein Authentifizierungs- und Zugriffssteuersegment enthält.

9. Verfahren, wie in Anspruch 1 vorgetragen, wobei der ausgewählte Benutzeridentifikator und das entsprechende private Schlüsselwort automatisch zu einem oder mehr Diensthostcomputern (**206**) übergeben werden.

10. Verfahren, wie in Anspruch 1 vorgetragen, ferner umfassend Anzeigen der Liste von Diensten in einer Benutzerschnittstelle, angezeigt auf dem Administrationsclientcomputer (**216**).

11. Verfahren, wie in Anspruch 1 vorgetragen, ferner umfassend Aufbauen eines Dienstlokators durch das Managementkonsolenprogramm (**210**) zum Lokalisieren eines Dienstes auf einem Hostservercomputer (**206**).

12. Verfahren, wie in Anspruch 1 vorgetragen, wobei Transferieren von einer oder mehr Managementdateien auf dem Hostserver (**206**) zu dem Administrationsserver (**208**) ferner Initiieren einer gemeinsamen Gateway-Schnittstelle (**226**) auf dem Administrationsservercomputer umfasst, wobei dadurch der Transfer von einer oder mehr Managementdateien und einer Vielzahl von Betriebssystembefehlen ermöglicht wird.

13. System zum Sichern einer Administration von Diensten (**218**), die sich auf einem oder mehr Diensthostcomputern (**206**) befinden, von einem Administrationsservercomputer (**208**), wobei der Administrationsservercomputer (**208**) mit einem Administrationsclient (**216**), der ein Programm eines Browsertyps hat, und mit dem einen oder mehr Diensthostcomputern (**206**) unter Verwendung eines Internetprotokolls verbunden ist, das System umfassend:
eine Benutzerprofildatenablage (**212**) zum Speichern von Daten bezüglich Benutzerprivilegien, wobei die Daten für jeden Benutzer einen Benutzerzugriffsgrad, eine Liste von Diensten und ein Passwort inkludieren;
eine Dienstmanagerteilkomponente (**210**) einer Kommunikationsschnittstelle, die sich auf dem Administrationsservercomputer (**208**) befindet, zum Akzeptieren eines Benutzeridentifikators und eines entsprechenden Schlüsselwortes und Übergeben des Benutzeridentifikators und des entsprechenden Schlüsselwortes zu der Benutzerprofildatenablage (**212**);
ein Komponentenkonfigurationsverzeichnis, geeignet, sich auf dem einen oder mehr Diensthosts (**206**) zu befinden, enthaltend Komponentenkonfigurationsdateien zum Speichern von Managementmodulen, die mit der Vielzahl von Diensten (**218**) in Verbindung stehen, wobei die Managementmodule Managementdaten enthalten, die beim Administrieren der Vielzahl von Diensten (**218**) genutzt werden;
eine Diensthostteilkomponente der Kommunikationsschnittstelle, die sich auf dem Administrationsservercomputer (**208**) befindet, zum Akzeptieren des Benutzeridentifikators und des entsprechenden Schlüsselwortes und Übergeben des Benutzeridentifikators und des entsprechenden Schlüsselwortes zu der Vielzahl von Diensthostcomputern (**206**) zum Verifizieren durch Untersuchen von Daten bezüglich Benutzerprivilegien, die in der Benutzerprofildatenablage (**212**) gespeichert sind.

14. System zum Sichern von Zugriff auf die Administration einer Vielzahl von verschiedenen Diensten (**218**), die sich auf einem oder mehr Diensthostcomputern (**206**) befinden, von einem Administrationsservercomputer (**208**), der mit dem einen oder mehr Diensthostcomputern (**206**) und mit einem Administrationsclientcomputer (**216**) verbunden ist, wobei es einen Dienstmanager (**210**) gibt, der sich auf dem Administrationsservercomputer (**208**) befindet, das System umfassend:
eine Kommunikationsverbindung zwischen dem Administrationsclientcomputer und dem Administrationsservercomputer, die zum Vorsehen eines ausgewählten Benutzeridentifikators und eines entsprechenden privaten Schlüsselwortes zu dem Dienstmanager (**210**) verwendet werden, wobei der Benutzeridentifikator angeordnet ist, einen Benutzer mit Administrationszugriff auf mindestens einen der Dienste (**218**) zu identifizieren;
eine Authentifizierungseinrichtung, konfiguriert zum Authentifizieren des Benutzers, durch Vergleichen des ausgewählten Benutzeridentifikators und des entsprechenden privaten Schlüsselwortes gegen eine Vielzahl von Benutzeridentifikatoren und privaten Schlüsselwörtern, die in einem persistenten Speicherbereich (**212**) gespeichert sind, wobei der Vergleich unter Steuerung des Dienstmanagers (**210**) durchgeführt wird;

einen Zugriffssteuermechanismus zum Ableiten einer Liste von Diensten, auf die der Benutzer, der mit dem Benutzeridentifikator in Verbindung steht, administrativen Zugriff hat;
eine Diensthostverifizierungseinrichtung zum Verifizieren, dass dem Benutzer, der mit dem ausgewählten Benutzeridentifikator in Verbindung steht, gestattet ist, auf einen ausgewählten der Dienste (**218**) in der abgeleiteten Liste von Diensten zuzugreifen, wobei sich die Verifizierungseinrichtung in dem Diensthostcomputer (**206**) befindet, der mit dem ausgewählten Dienst in Verbindung steht, und Zugriffssteuerdaten nutzt, die mit dem ausgewählten Benutzeridentifikator in dem persistenten Speicherbereich (**212**) in Verbindung stehen; und
eine Datentransferkomponente zum Transferieren von einer oder mehr Managementdateien auf dem Diensthostcomputer (**206**) zu dem Administrationsservercomputer (**208**), wobei dadurch eine Manipulation der Managementdateien unter Nutzung des Dienstmanagers (**210**) erleichtert wird.

15. Computerlesbares Medium, konfiguriert, Computerprogrammierinstruktionen zum Sichern eines Zugriffs auf die Administration einer Vielzahl von verschiedenen Diensten (**218**), die sich auf einem oder mehr Diensthostcomputern (**218**) befinden, von einem Administrationsservercomputer (**208**), der mit dem einen oder mehr Diensthostcomputern (**206**) in Verbindung steht, zu speichern, wobei es einen Dienstmanager (**210**) gibt, der sich auf dem Administrationsservercomputer (**208**) befindet, das computerlesbare Medium umfassend:
Computerprogrammierinstruktionen zum Vorsehen eines ausgewählten Benutzeridentifikators und eines entsprechenden privaten Schlüsselwortes, wobei der Benutzeridentifikator angeordnet ist, einen Benutzer mit administrativem Zugriff auf mindestens einen der verschiedenen Dienste (**218**) zu identifizieren;
Computerprogrammierinstruktionen zum Authentifizieren des Benutzers durch Vergleichen des ausgewählten Benutzeridentifikators und des entsprechenden privaten Schlüsselwortes gegen eine Vielzahl von Benutzeridentifikatoren und privaten Schlüsselwörtern, die in einem persistenten Speicherbereich (**212**) gespeichert sind, wobei der Vergleich unter Steuerung des Dienstmanagers (**210**) durchgeführt wird;
Computerprogrammierinstruktionen zum Ableiten einer Liste von Diensten, auf die der Benutzer, der mit dem Benutzeridentifikator in Verbindung steht, administrativen Zugriff hat;
wenn eine Anfrage durchgeführt wird, einen ausgewählten der Dienste in der abgeleiteten Liste von Diensten zu administrieren, Computerprogrammierinstruktionen zum Verifizieren in dem Diensthostcomputer (**206**), der mit dem ausgewählten Dienst in Verbindung steht, dass dem Benutzer, der mit dem ausgewählten Benutzeridentifikator in Verbindung steht, gestattet ist, auf den ausgewählten Dienst zuzugreifen, durch Untersuchen von Zugriffssteuerdaten, die mit dem ausgewählten Benutzeridentifikator in dem persistenten Speicherbereich (**212**) in Verbindung stehen, und
Computerprogrammierinstruktionen zum Transferieren von einer oder mehr Managementdateien auf dem Diensthostcomputer (**206**) zu dem Administrationsserver (**208**), wobei dadurch eine Manipulation der Managementdateien unter Nutzung des Dienstmanagers (**210**) erleichtert wird.

16. Computerlesbares Medium, wie in Anspruch 15 vorgetragen, wobei der Administrationsservercomputer mit einem Administrationsclientcomputer verbunden ist, der geeignet ist, ein Browserprogramm ablaufen zu lassen, und wobei der ausgewählte Benutzeridentifikator und das entsprechende private Schlüsselwort über eine Kommunikationsverbindung zwischen dem Administrationsclientcomputer und dem Administrationsservercomputer vorgesehen werden, wobei die Kommunikationsverbindungen zwischen dem Administrationsservercomputer, dem Administrationsclientcomputer und dem einen oder mehr Diensthostcomputern ein Internetprotokoll nutzen.

17. Computerlesbares Medium, wie in Anspruch 15 vorgetragen, wobei das Verifizieren in dem Diensthostcomputer, dass dem Benutzer, der mit dem ausgewählten Benutzeridentifikator in Verbindung steht, gestattet ist, auf den ausgewählten Dienst von der Liste von Diensten zuzugreifen, ferner Kommunizieren des ausgewählten Benutzeridentifikators und des entsprechenden privaten Schlüsselwortes zu dem Hostservercomputer unter Verwendung einer gemeinsamen Gateway-Schnittstelle umfasst.

18. Computerlesbares Medium, wie in Anspruch 15 vorgetragen, wobei das Transferieren von einer oder mehr Managementdateien auf dem Hostserver zu dem Administrationsserver ferner Initiieren einer gemeinsamen Gateway-Schnittstelle auf dem Administrationsservercomputer umfasst, wobei dadurch der Transfer von einer oder mehr Managementdateien und einer Vielzahl von Betriebssystembefehlen ermöglicht wird.

19. System, wie in Anspruch 14 vorgetragen, wobei das Transferieren von einer oder mehr Managementdateien auf dem Host-Server zu dem Administrationsserver ferner Initiieren einer gemeinsamen Gateway-Schnittstelle auf dem Administrationsservercomputer umfasst, wobei dadurch der Transfer von einer oder mehr Managementdateien und einer Vielzahl von Betriebssystembefehlen ermöglicht wird.

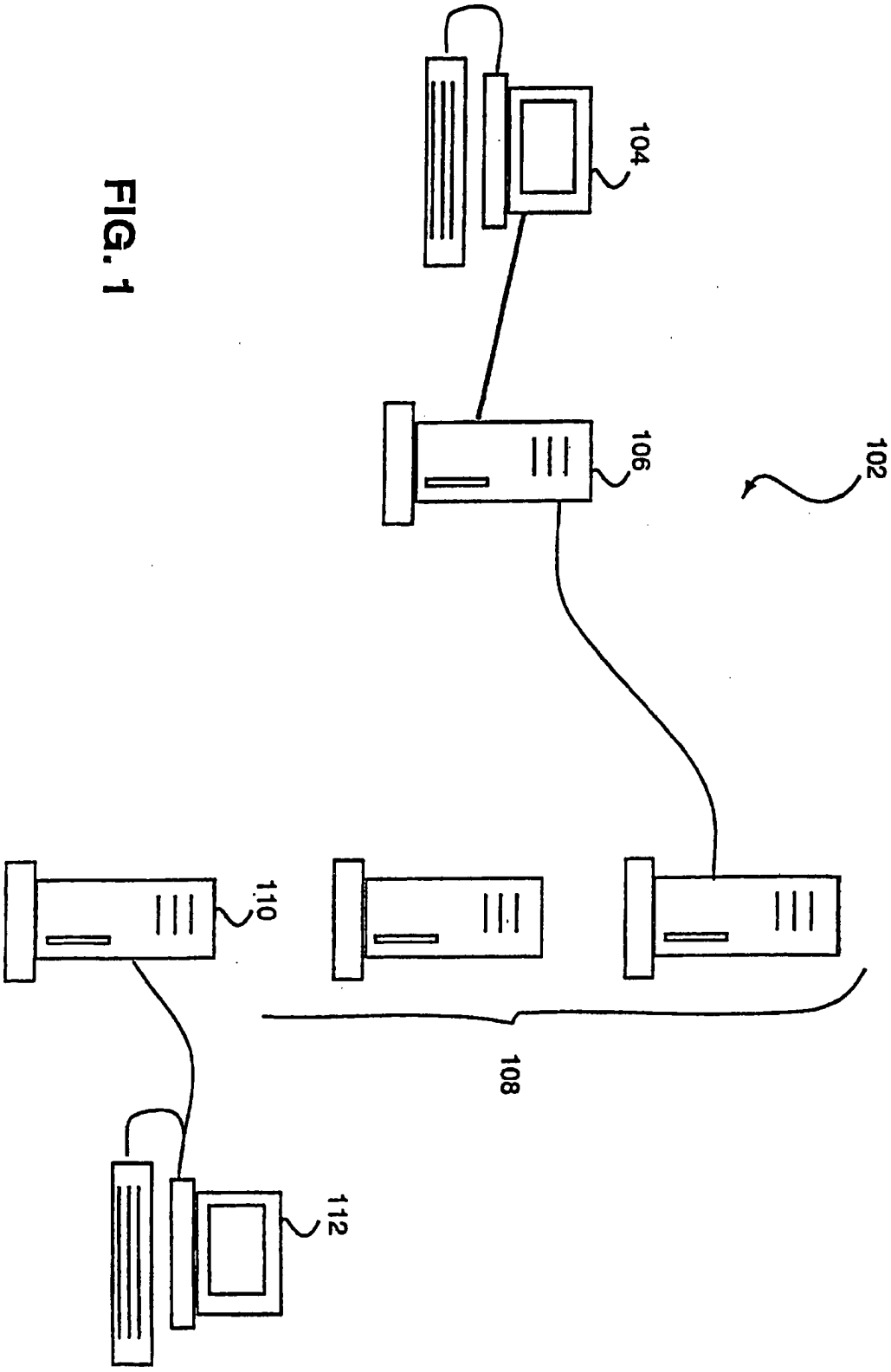
20. System, wie in Anspruch 14 vorgetragen, wobei das Verifizieren auf dem Diensthostcomputer, dass dem Benutzer, der mit dem ausgewählten Be-

nutzeridentifikator in Verbindung steht, gestattet ist, auf den ausgewählten Dienst von der Liste von Diensten zuzugreifen, ferner Kommunizieren des ausgewählten Benutzeridentifikators und des entsprechenden privaten Schlüsselwortes zu dem Hostservercomputer unter Verwendung einer gemeinsamen Gateway-Schnittstelle umfasst.

Es folgen 12 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1



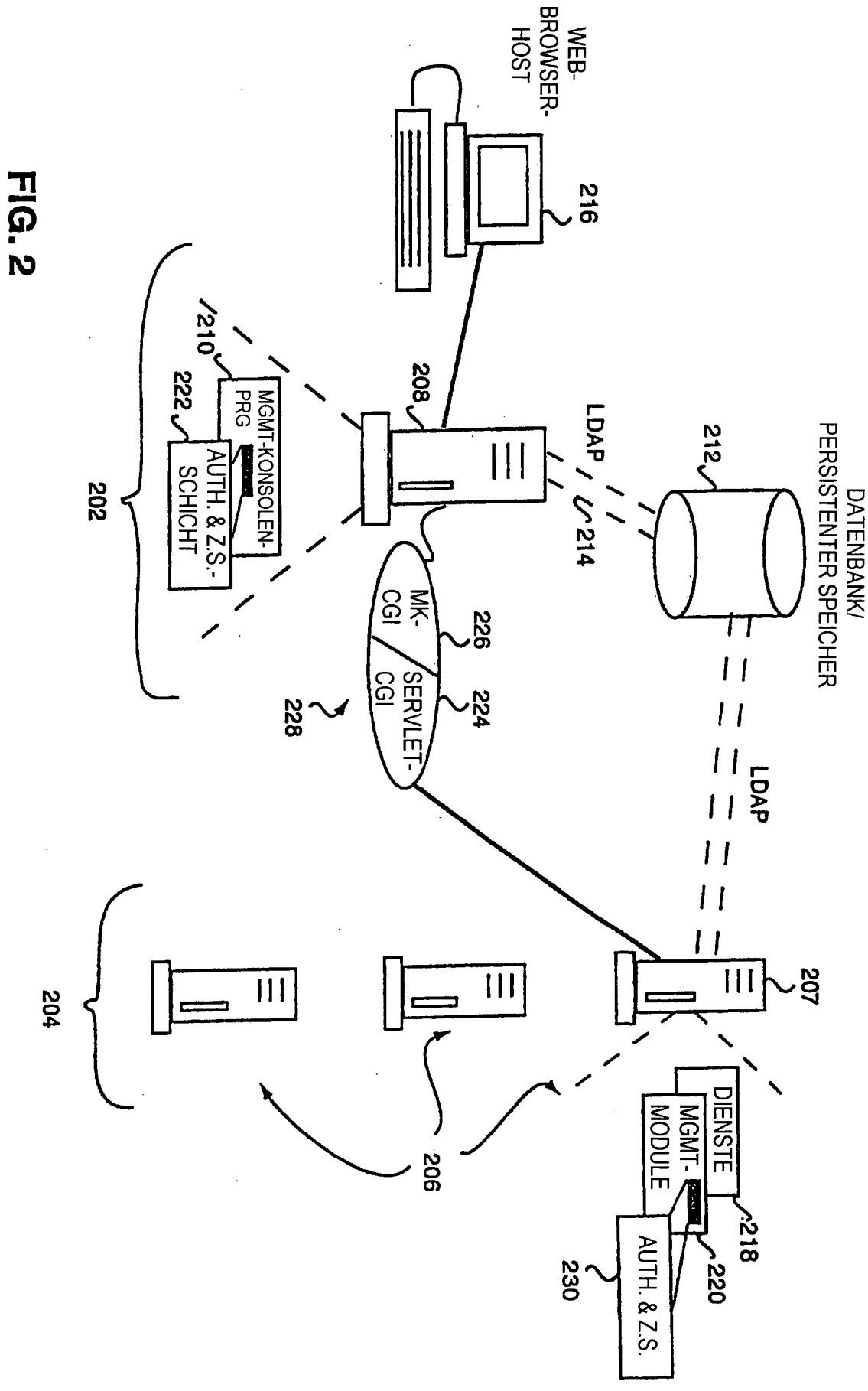

WEB-
BROWSER-
HOST
216
208
LDAP
214
212
DATENBANK/
PERSISTENTER SPEICHER
LDAP
210
MGMT-KONSOLEN-
PRG
222
AUTH. & Z.S.-
SCHICHT
202
MK-
CGI
226
SERVLET-
CGI
224
228
207
DIENSTE
MGMT-
MODULE
218
220
AUTH. & Z.S.
230
204
206

FIG. 2

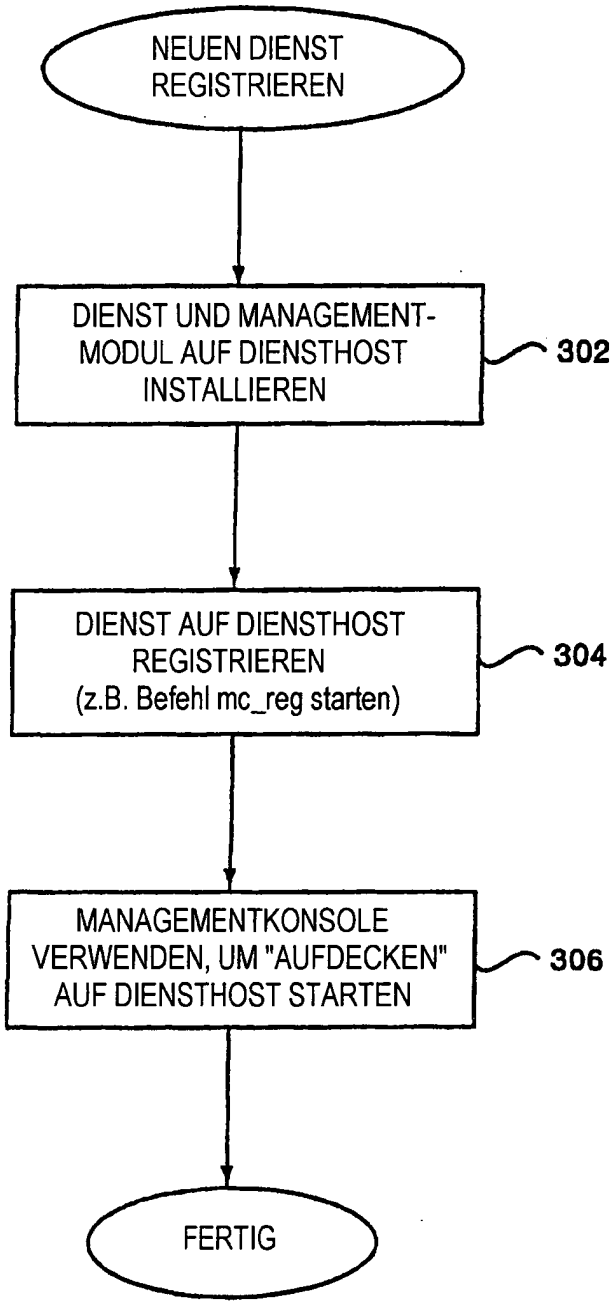

NEUEN DIENST REGISTRIEREN
DIENST UND MANAGEMENT-MODUL AUF DIENSTHOST INSTALLIEREN
302
DIENST AUF DIENSTHOST REGISTRIEREN (z.B. Befehl mc_reg starten)
304
MANAGEMENTKONSOLE VERWENDEN, UM "AUFDECKEN" AUF DIENSTHOST STARTEN
306
FERTIG

FIG. 3

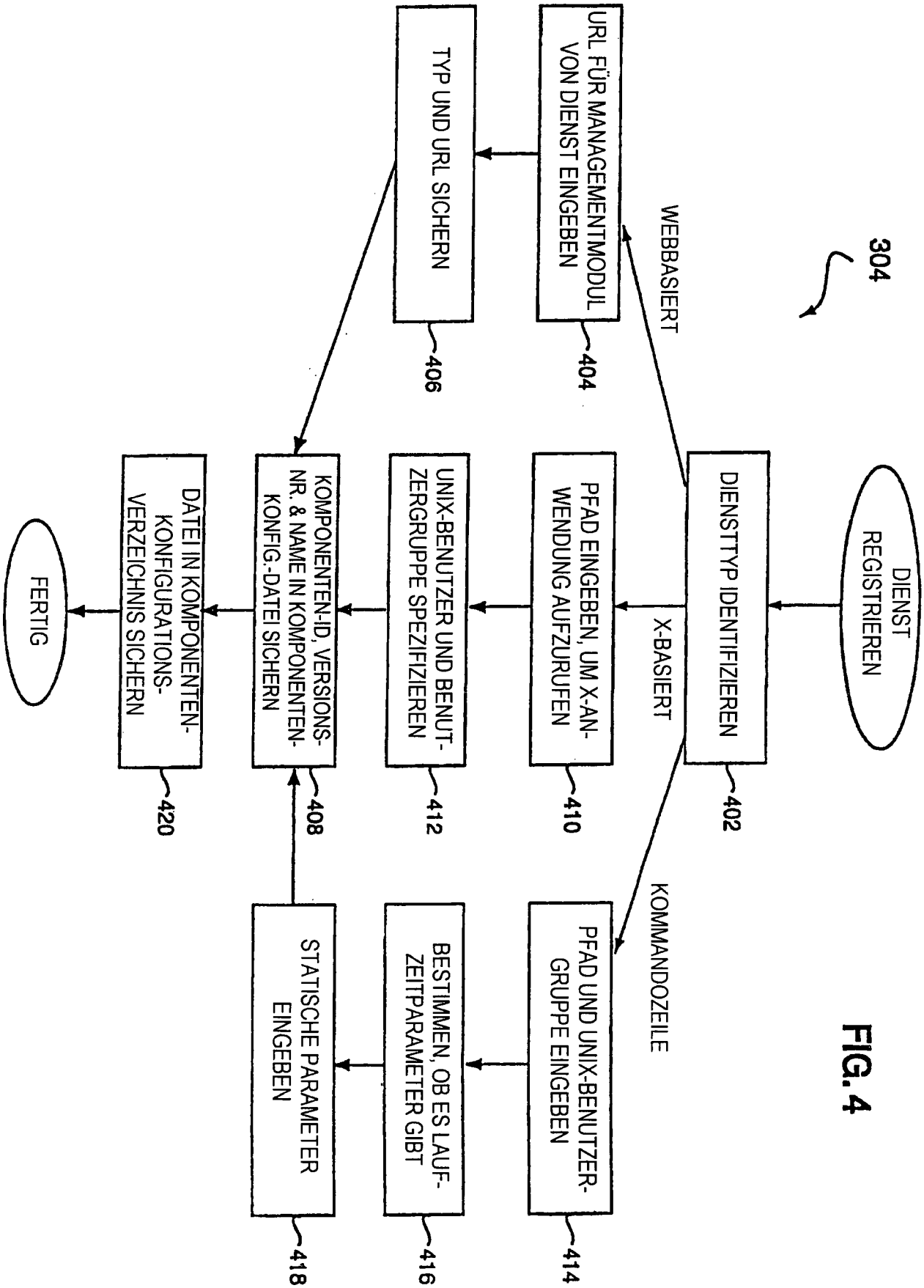

304
DIENST REGISTRIEREN
DIENSTTYP IDENTIFIZIEREN
402
WEBBASIERT
X-BASIERT
KOMMANDOZEILE
URL FÜR MANAGEMENTMODUL VON DIENST EINGEBEN
404
TYP UND URL SICHERN
406
PFAD EINGEBEN, UM X-ANWENDUNG AUFZURUFEN
410
UNIX-BENUTZER UND BENUTZERGRUPPE SPEZIFIZIEREN
412
KOMPONENTEN-ID, VERSIONS-NR. & NAME IN KOMPONENTEN-KONFIG.-DATEI SICHERN
408
DATEI IN KOMPONENTEN-KONFIGURATIONS-VERZEICHNIS SICHERN
420
FERTIG
PFAD UND UNIX-BENUTZER-GRUPPE EINGEBEN
414
BESTIMMEN, OB ES LAUF-ZEITPARAMETER GIBT
416
STATISCHE PARAMETER EINGEBEN
418

FIG. 4

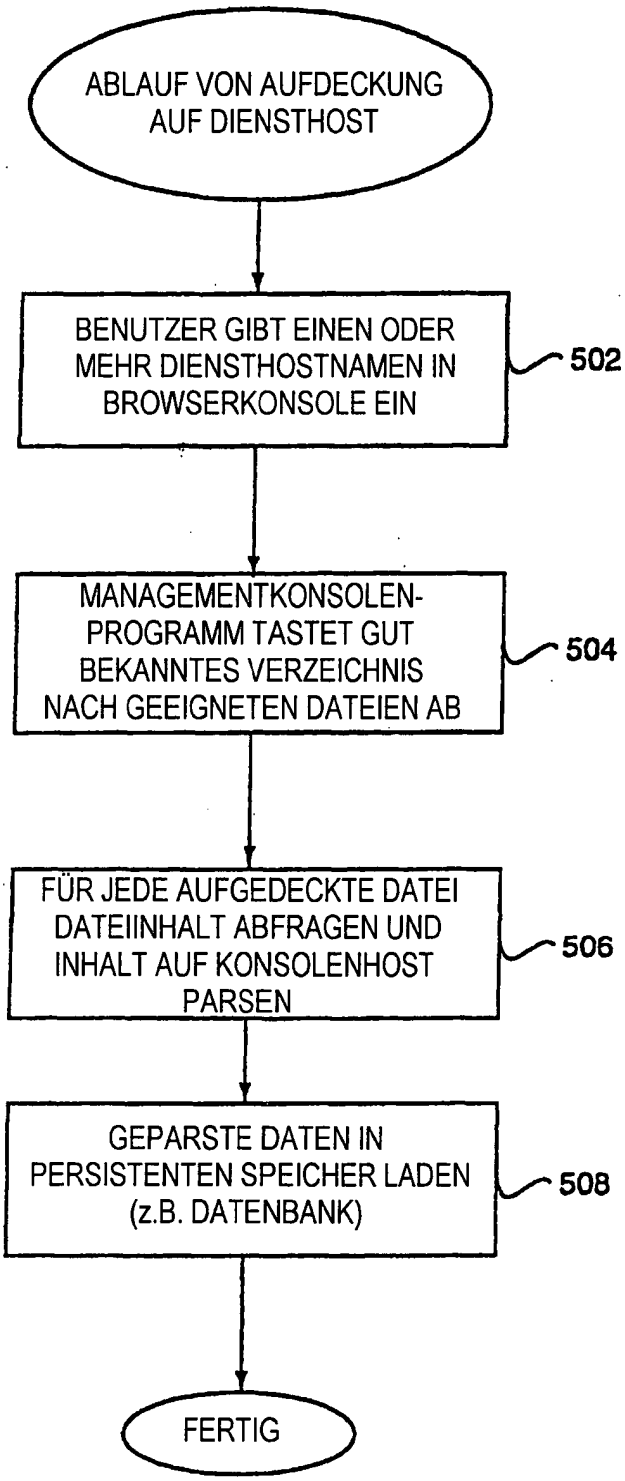
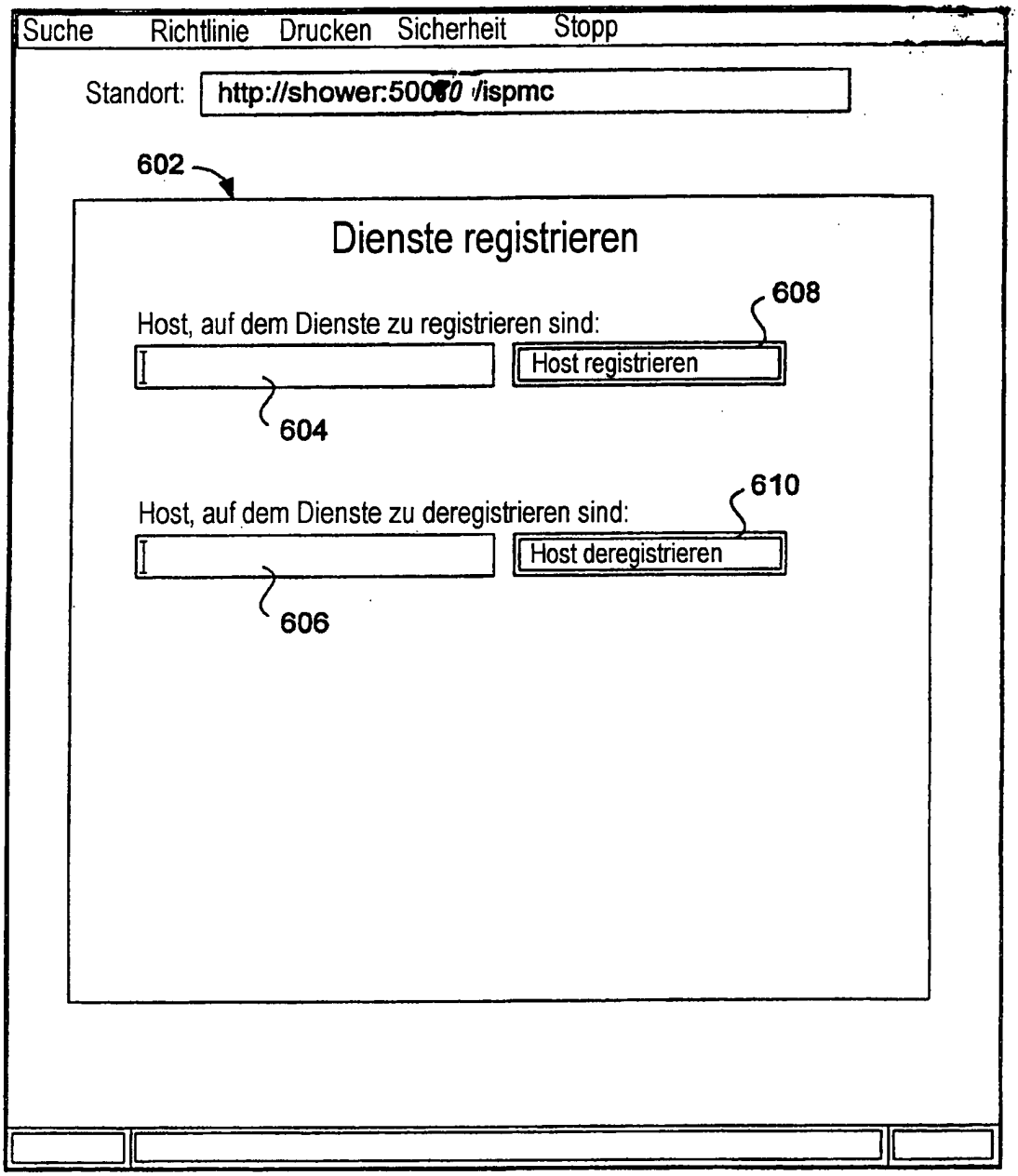
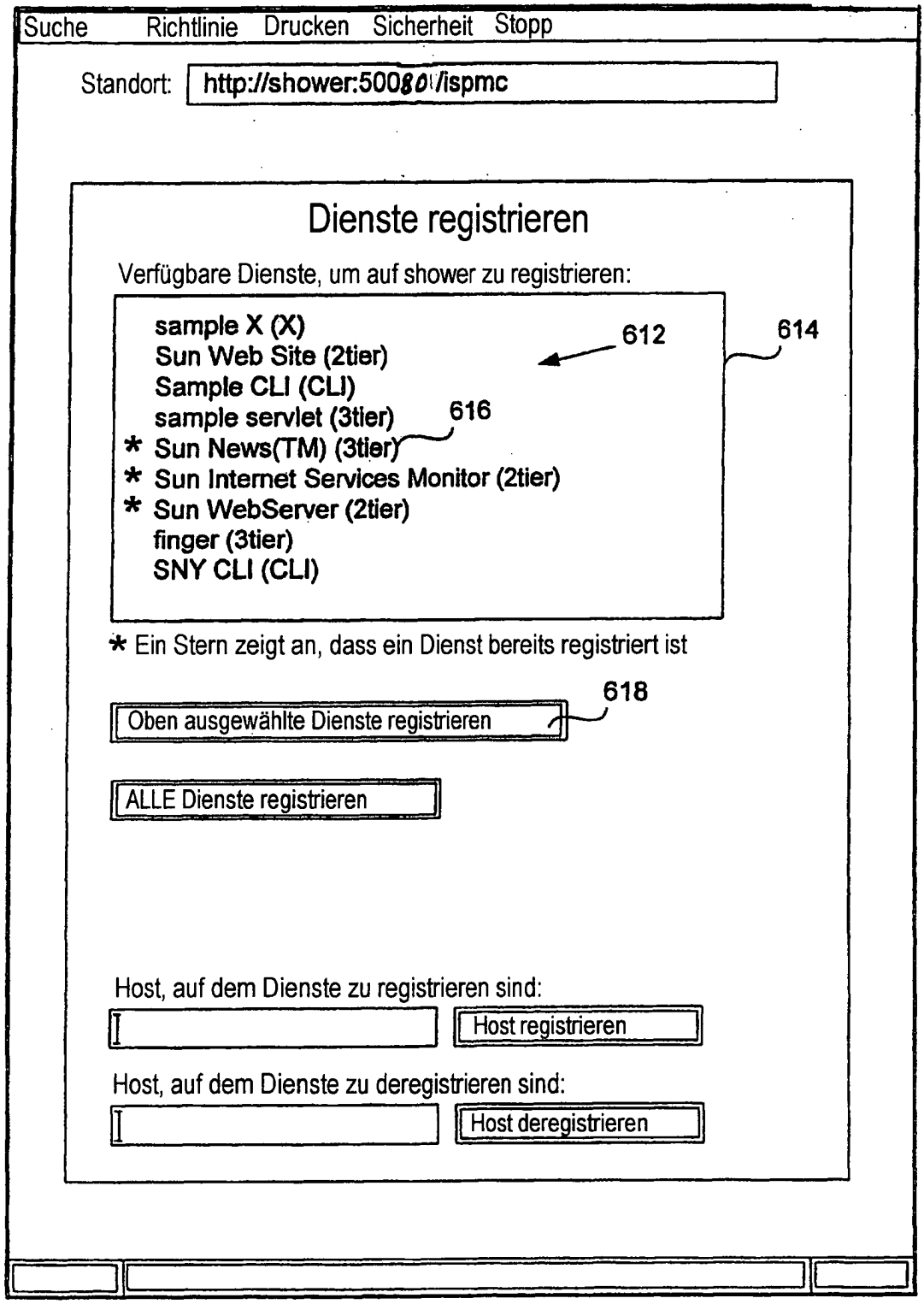

ABLAUF VON AUFDECKUNG AUF DIENSTHOST
BENUTZER GIBT EINEN ODER MEHR DIENSTHOSTNAMEN IN BROWSERKONSOLE EIN
502
MANAGEMENTKONSOLEN-PROGRAMM TASTET GUT BEKANNTES VERZEICHNIS NACH GEEIGNETEN DATEIEN AB
504
FÜR JEDE AUFGEDECKTE DATEI DATEIINHALT ABFRAGEN UND INHALT AUF KONSOLENHOST PARSEN
506
GEPARSTE DATEN IN PERSISTENTEN SPEICHER LADEN (z.B. DATENBANK)
508
FERTIG

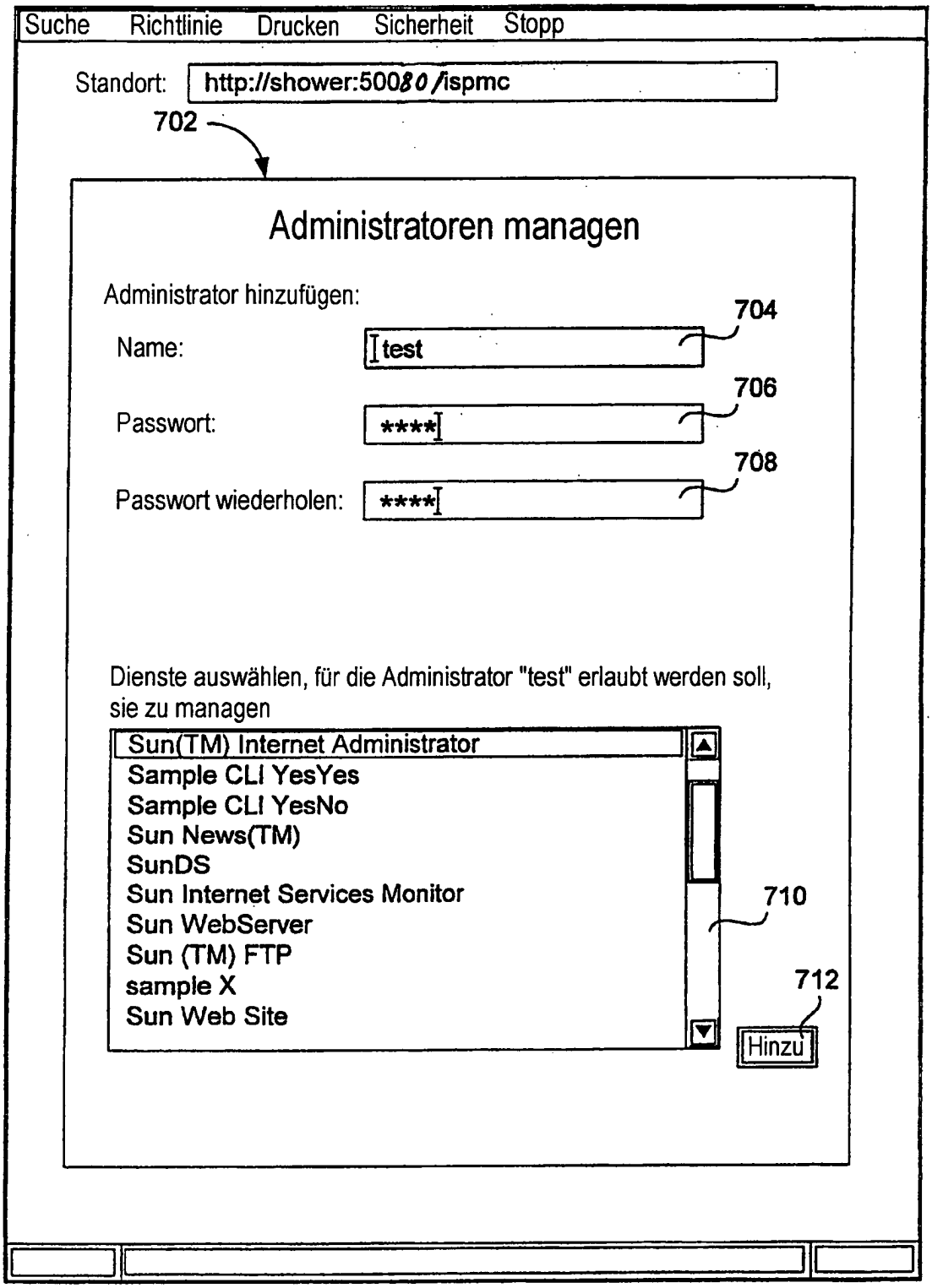
FIG. 5


Suche
Richtlinie
Drucken
Sicherheit
Stopp
Standort:
http://shower:5006 /ispmc
602
Dienste registrieren
608
Host, auf dem Dienste zu registrieren sind:
Host registrieren
604
610
Host, auf dem Dienste zu deregistrieren sind:
Host deregistrieren
606

Figur 6A


Suche Richtlinie Drucken Sicherheit Stopp
Standort: http://shower:50080/ispmc
Dienste registrieren
Verfügbare Dienste, um auf shower zu registrieren:
sample X (X)
Sun Web Site (2tier)
Sample CLI (CLI)
sample servlet (3tier)
* Sun News(TM) (3tier)
* Sun Internet Services Monitor (2tier)
* Sun WebServer (2tier)
finger (3tier)
SNY CLI (CLI)
612
614
616
* Ein Stern zeigt an, dass ein Dienst bereits registriert ist
618
Oben ausgewählte Dienste registrieren
ALLE Dienste registrieren
Host, auf dem Dienste zu registrieren sind:
Host registrieren
Host, auf dem Dienste zu deregistrieren sind:
Host deregistrieren

Figur 6B


Suche
Richtlinie
Drucken
Sicherheit
Stopp
Standort:
http://shower:50080/ispmc
702
Administratoren managen
Administrator hinzufügen:
Name:
test
704
Passwort:
****
706
Passwort wiederholen:
****
708
Dienste auswählen, für die Administrator "test" erlaubt werden soll, sie zu managen
Sun(TM) Internet Administrator
Sample CLI YesYes
Sample CLI YesNo
Sun News(TM)
SunDS
Sun Internet Services Monitor
Sun WebServer
Sun (TM) FTP
sample X
Sun Web Site
710
712
Hinzu

Figur 7

FIG. 8a

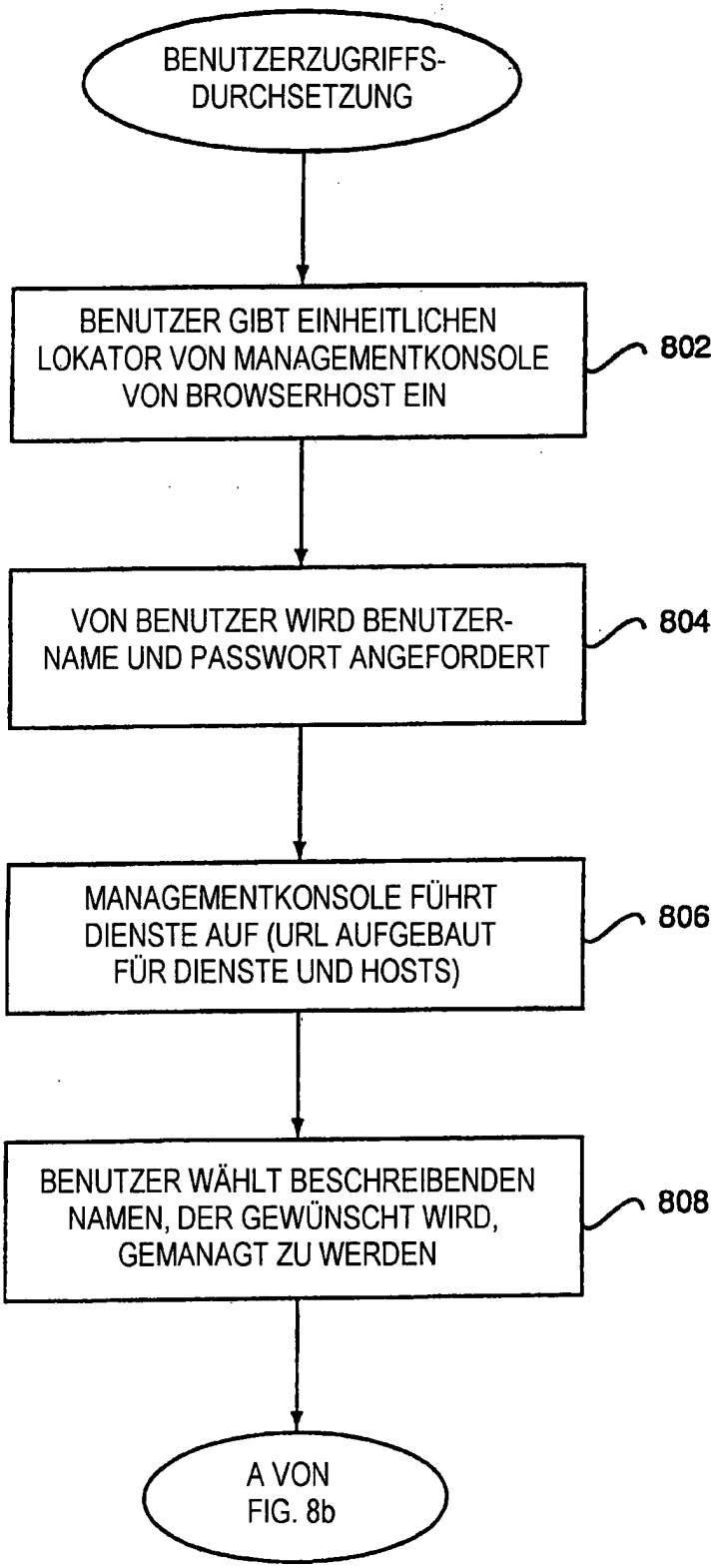

BENUTZERZUGRIFFS-
DURCHSETZUNG
BENUTZER GIBT EINHEITLICHEN LOKATOR VON MANAGEMENTKONSOLE VON BROWSERHOST EIN
802
VON BENUTZER WIRD BENUTZER-NAME UND PASSWORT ANGEFORDERT
804
MANAGEMENTKONSOLE FÜHRT DIENSTE AUF (URL AUFGEBAUT FÜR DIENSTE UND HOSTS)
806
BENUTZER WÄHLT BESCHREIBENDEN NAMEN, DER GEWÜNSCHT WIRD, GEMANAGT ZU WERDEN
808
A VON
FIG. 8b

FIG. 8b

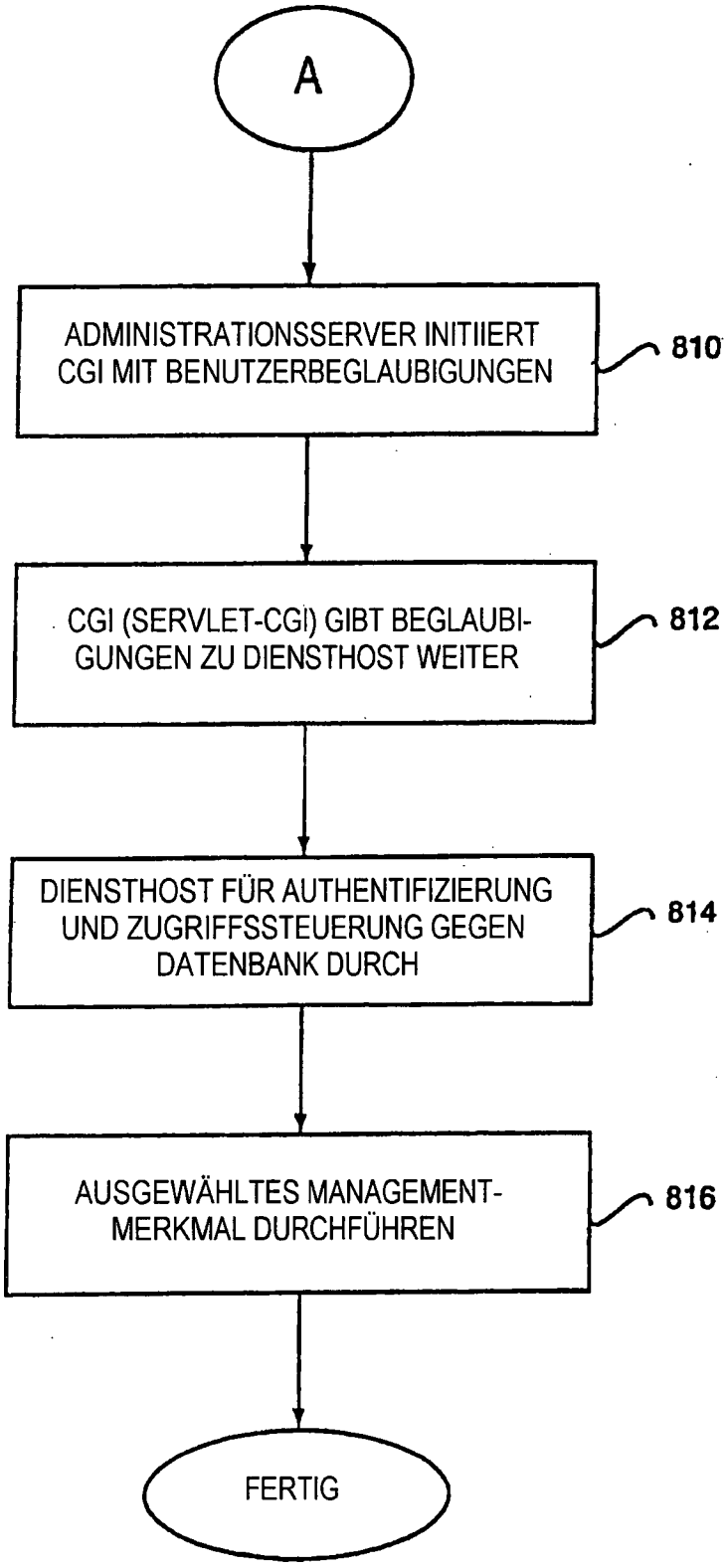
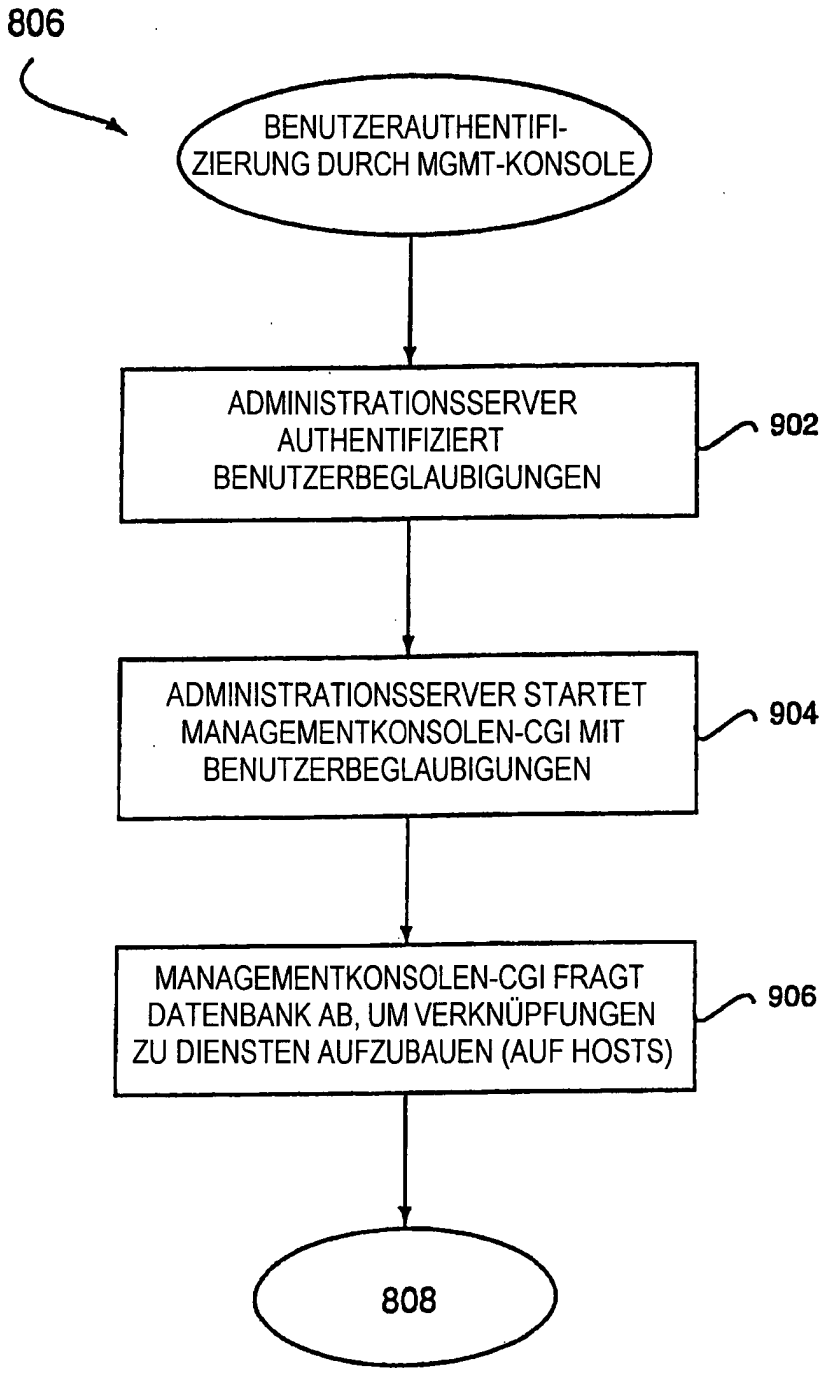

A
ADMINISTRATIONSSERVER INITIIERT CGI MIT BENUTZERBEGLAUBIGUNGEN
810
CGI (SERVLET-CGI) GIBT BEGLAUBIGUNGEN ZU DIENSTHOST WEITER
812
DIENSTHOST FÜR AUTHENTIFIZIERUNG UND ZUGRIFFSSTEUERUNG GEGEN DATENBANK DURCH
814
AUSGEWÄHLTES MANAGEMENT-MERKMAL DURCHFÜHREN
816
FERTIG

FIG. 9


806
BENUTZERAUTHENTIFI-
ZIERUNG DURCH MGMT-KONSOLE
ADMINISTRATIONSSERVER
AUTHENTIFIZIERT
BENUTZERBEGLAUBIGUNGEN
902
ADMINISTRATIONSSERVER STARTET
MANAGEMENTKONSOLEN-CGI MIT
BENUTZERBEGLAUBIGUNGEN
904
MANAGEMENTKONSOLEN-CGI FRAGT
DATENBANK AB, UM VERKNÜPFUNGEN
ZU DIENSTEN AUFZUBAUEN (AUF HOSTS)
906
808

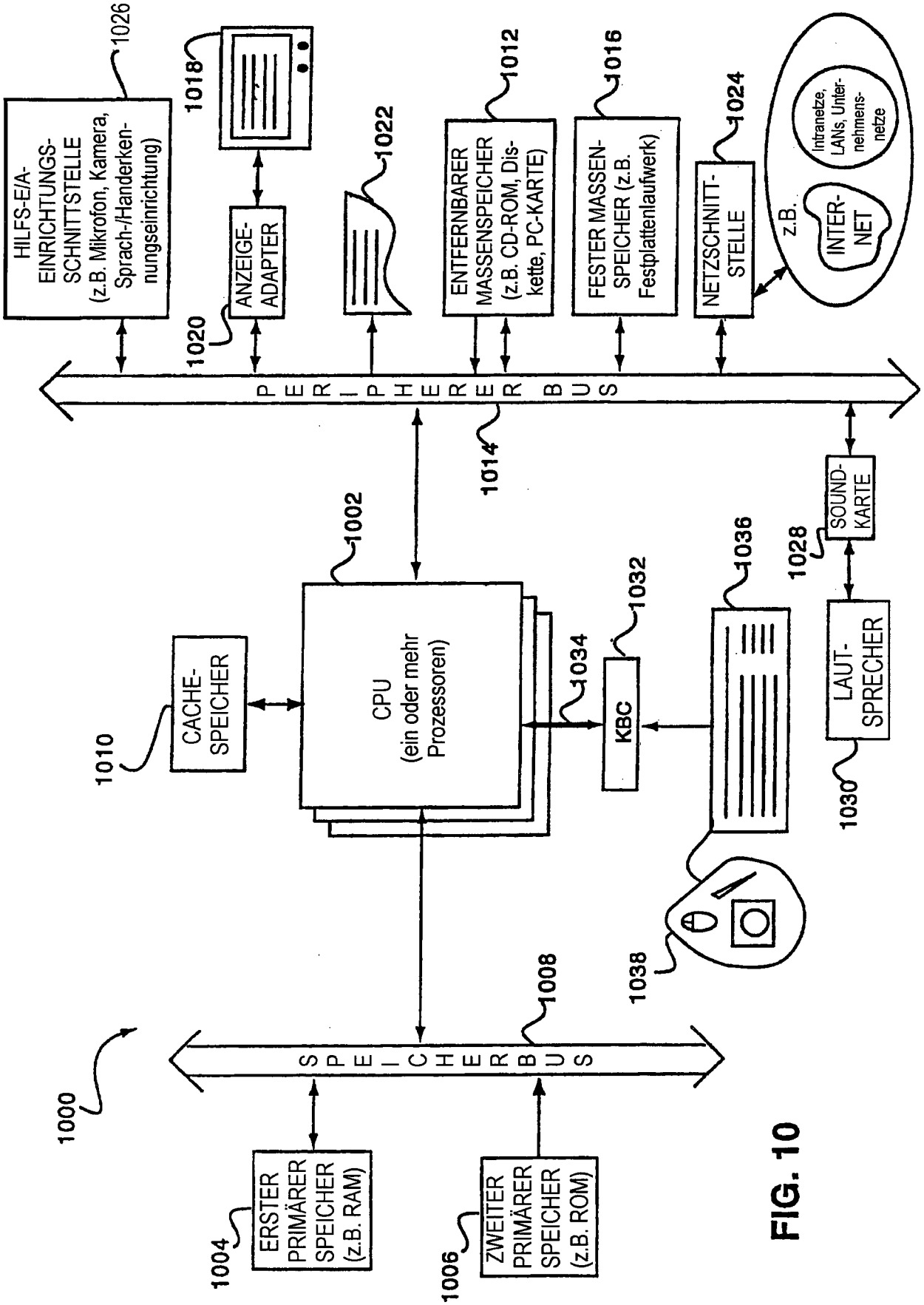

1000
1002
1004
1006
1008
1010
1012
1014
1016
1018
1020
1022
1024
1026
1028
1030
1032
1034
1036
1038
HILFS-E/A-EINRICHTUNGS-SCHNITTSTELLE (z.B. Mikrofon, Kamera, Sprach-/Handerkennungseinrichtung)
ANZEIGE-ADAPTER
ENTFERNBARER MASSENSPEICHER (z.B. CD-ROM, Diskette, PC-KARTE)
FESTER MASSEN-SPEICHER (z.B. Festplattenlaufwerk)
NETZSCHNITT-STELLE
z.B.
INTER-NET
Intranetze, LANs, Unternehmensnetze
PERIPHERER BUS
SOUND-KARTE
LAUT-SPRECHER
CACHE-SPEICHER
CPU (ein oder mehr Prozessoren)
KBC
SPEICHER BUS
ERSTER PRIMÄRER SPEICHER (z.B. RAM)
ZWEITER PRIMÄRER SPEICHER (z.B. ROM)

FIG. 10