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

(10) **DE 697 31 069 T2** 2005.02.17

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 862 301 B1**
(21) Deutsches Aktenzeichen: **697 31 069.8**
(96) Europäisches Aktenzeichen: **97 305 086.7**
(96) Europäischer Anmeldetag: **10.07.1997**
(97) Erstveröffentlichung durch das EPA: **02.09.1998**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **06.10.2004**
(47) Veröffentlichungstag im Patentblatt: **17.02.2005**

(51) Int Cl.[7]: **H04L 29/06**
G06F 9/445

(30) Unionspriorität:
31750596 28.11.1996 JP

(84) Benannte Vertragsstaaten:
DE, FR, GB

(73) Patentinhaber:
Fujitsu Ltd., Kawasaki, Kanagawa, JP

(74) Vertreter:
W. Seeger und Kollegen, 81369 München

(72) Erfinder:
Saito, Tamio, Nakahara-ku, Kanagawa 211, JP;
Tokuyo, Masanaga, Nakahara-ku, Kanagawa 211, JP;
Iida, Ichiro, Nakahara-ku, Kanagawa 211, JP;
Takada, Yuji, Nakahara-ku, Kanagawa 211, JP

(54) Bezeichnung: **Verschlüsselungskommunikationsystem mit einem Agent und einem Speichermedium zur Speicherung dieses Agents**

Beschreibung

[0001] Diese Erfindung bezieht sich auf ein Verschlüsselungskommunikationsverfahren, um den Diebstahl und das Abfangen sowie Verfälschen von Informationen in einer Kommunikation zwischen Computern zu verhindern; insbesondere bezieht sie sich auf ein Verfahren zur Verschlüsselung unter Verwendung eines Agenten.

[0002] In den letzten Jahren hat mit der explosionsartigen Verbreitung von Internet und Intranets die Bedeutung der Informationssicherheit ständig zugenommen. Bekannte Sicherheitsverfahren in der Informationskommunikation enthalten das Verfahren zur Verschlüsselung von Informationen unter Verwendung von Chiffrierschlüsseln, die Terminals gemeinsam besitzen, wie der DES (Data Encryption Standard), und ein Verschlüsselungsverfahren wie RSA, bei dem Terminals öffentliche Schlüssel austauschen, und Informationen unter Verwendung ihrer privaten Schlüssel verschlüsselt werden. Zusätzlich gibt es zur Erhöhung der Verschlüsselungssicherheit bekannte Verfahren, wie die Änderung des Chiffrierschlüssels und des Verschlüsselungsverfahrens mit der Zeit, wie beispielsweise in Tokkaihei (Japanische Patentveröffentlichung) 1-212041 beschrieben.

[0003] In einem herkömmlichen gewöhnlichen Verschlüsselungskommunikationssystem ist das Verschlüsselungsverfahren öffentlich bekannt. Um ein starkes Kryptosystem zu erhalten, muss aus diesem Grund der bei der Verschlüsselung verwendete Schlüssel eine große Anzahl von Bits aufweisen. Wenn jedoch der bei der Verschlüsselung verwendete Schlüssel eine große Anzahl von Bits aufweist, wird die für die Verschlüsselungs- und Entschlüsselungsverarbeitung erforderliche Zeit unweigerlich lang. Insbesondere wenn die Verschlüsselung in Echtzeitanwendungen erfolgt (wie Sprache und Bilder), ist die Langsamkeit der Verarbeitung bei herkömmlichen Verfahren wie DES und RSA ein Problem.

[0004] Wenn die Verschlüsselung durch ein Verfahren erfolgt, das die Änderung der Kombination des Chiffrierschlüssels und des Verschlüsselungsverfahrens mit der Zeit involviert, ist es notwendig, den Chiffrierschlüssel vorher zu registrieren und das Verschlüsselungsprogramm in den Terminals vorher zu installieren, die für eine solche Verschlüsselungskommunikation verwendet werden. Jedesmal wenn neue Terminaleinrichtungen in einem Netz hinzugefügt werden, ist es demgemäß notwendig, den Chiffrierschlüssel zu registrieren und das Verschlüsselungsprogramm in diesem Terminals zu installieren.

[0005] Die US-4 984 271-A offenbart ein Verschlüsselungskommunikationsverfahren. Bei diesem Verfahren wird ein Chiffrierprozessprogramm von einem Host an ein Terminal gesendet, wenn das Terminal eingeschaltet wird, und wird gelöscht, wenn das Terminal ausgeschaltet wird.

[0006] Ein Zweck dieser Erfindung ist, ein starkes Verschlüsselungsverfahren vorzusehen, das eine adäquate Verarbeitungsgeschwindigkeit aufweist, so dass in der Praxis kein Problem beim Echtzeittransfer von Daten auftritt.

[0007] Ein weiterer Zweck dieser Erfindung ist zu ermöglichen, eine verschlüsselte Kommunikation zwischen Terminals durchzuführen, ohne dass dasselbe Verschlüsselungsprogramm in diesen vorinstalliert werden muss.

[0008] Die Erfindung wird mittels der Verfahren von Anspruch 1 und 2, der Speichermedien von Anspruch 12 und 13, des Systems von Anspruch 15 und der Vorrichtungen von Anspruch 16 und 17 durchgeführt.

[0009] Weitere Aspekte der Erfindung sind wie in den anderen unabhängigen Ansprüchen definiert.

[0010] Nun wird anhand bloßer Beispiele auf die beigeschlossenen Zeichnungen Bezug genommen, in denen: –

[0011] **Fig.** 1 eine Darstellung ist, welche eine Übersicht über diese Erfindung erläutert;

[0012] **Fig.** 2 eine Konfigurationsdarstellung des Servers und des Clients ist;

[0013] **Fig.** 3 eine Konfigurationsdarstellung eines vertrauenswürdigen Agenten ist, der in mobilem (übertragbarem oder maschinenunabhängigem) Code beschrieben ist;

[0014] **Fig.** 4A und 4B Sequenzdiagramme sind, welche die Verarbeitung erläutern, die beim Senden eines

vertrauenswürdigen Agenten und Herstellen eines verschlüsselten Kommunikationswegs involviert ist;

[0015] **Fig.** 5 eine Konfigurationsdarstellung des verschlüsselten Kommunikationssystems einer Ausführungsform dieser Erfindung ist;

[0016] **Fig.** 6 eine Konfigurationsdarstellung des verschlüsselten Kommunikationssystems einer weiteren Ausführungsform dieser Erfindung ist;

[0017] **Fig.** 7A und 7B Darstellungen sind, die ein Beispiel der Konfiguration der kryptographischen Verarbeitungseinheit zeigen;

[0018] **Fig.** 8 eine Darstellung ist, welche die Hardware-Schaltung einer Ausführungsform eines Pseudozufallszahlengenerators zeigt;

[0019] **Fig.** 9 eine Darstellung ist, welche die Konfiguration eines 3-stufigen Pseudozufallszahlengenerators zeigt, der die M-Serie generiert;

[0020] **Fig.** 10A bis 10C Figuren sind, die ein Beispiel eines Pseudozufallszahlen-Generierverfahrens zeigen;

[0021] **Fig.** 11 ein Flussdiagramm ist, das die Aktion der Verschlüsselungsverfahrensauswahl-Steuereinheit erläutert;

[0022] **Fig.** 12A ein Sequenzdiagramm ist, das die Verarbeitung auf der Sendeseite zeigt, wenn Daten zwischen Anwendungen gesendet und empfangen werden;

[0023] **Fig.** 12B eine Konfigurationsdarstellung eines Datenpakets ist, das chiffrierten Text sendet;

[0024] **Fig.** 13 ein Sequenzdiagramm ist, das die Verarbeitung auf der Empfangsseite zeigt, wenn Daten zwischen Anwendungen gesendet und empfangen werden;

[0025] **Fig.** 14A und 14B Darstellungen sind, die ein Verfahren zur Herstellung einer kryptographischen Synchronisation erläutern;

[0026] **Fig.** 15 eine Darstellung ist, die ein Beispiel der Konfiguration eines verschlüsselten Kommunikationssystems zeigt, das den Startparameter zum Zweck der Generierung von Pseudozufallszahlen ändert;

[0027] **Fig.** 16 eine Konfigurationsdarstellung eines verschlüsselten Kommunikationssystems einer weiteren Ausführungsform dieser Erfindung ist;

[0028] **Fig.** 17 eine Konfigurationsdarstellung eines verschlüsselten Kommunikationssystems noch einer weiteren Ausführungsform dieser Erfindung ist;

[0029] **Fig.** 18 ein Flussdiagramm ist, das die Aktion der Startparametergeneriersektion erläutert;

[0030] **Fig.** 19 eine Konfigurationsdarstellung für den Fall ist, in dem das verschlüsselte Kommunikationssystem dieser Ausführungsform an das WWW angepasst ist;

[0031] **Fig.** 20 eine Konfigurationsdarstellung für den Fall ist, in dem die verschlüsselte Kommunikation dieser Erfindung bei einem Bildtransmissions/Sprachtransmissionssystem angewendet wird;

[0032] **Fig.** 21 eine Konfigurationsdarstellung für den Fall ist, in dem die verschlüsselte Kommunikation dieser Erfindung bei einem elektronischen Konferenzsystem angewendet wird;

[0033] **Fig.** 22 eine Darstellung ist, welche die Verarbeitung zur Änderung des Kommunikationsobjekts eines Anwendungsprogramms erläutert, wenn ein vertrauenswürdiger Agent verwendet wird;

[0034] **Fig.** 23 bis 29 Figuren sind, die ein Beispiel eines Agentenprogramms zeigen, das zum Senden in mobilem Code beschriebener verschlüsselter Bilder verwendet wird.

[0035] In dem verschlüsselten Kommunikationssystem einer Ausführungsform dieser Erfindung wird zuerst

ein Agent zum Zweck der Verschlüsselungsverarbeitung in dem Sendeterminal installiert. Vor dem Transferieren von Daten sendet das Sendeterminal einen Agenten mit derselben Funktion wie der installierte Agent an das Empfangsterminal. Der Agent, der an das Empfangsterminal gesendet wird, ist in mobilem Code beschrieben. Wenn Daten transferiert werden, werden die Daten unter Verwendung dieses Agenten in dem Sendeterminal verschlüsselt, und in dem Empfangsterminal unter Verwendung des Agenten entschlüsselt, der vom Sendeterminal gesendet wurde.

[0036] Somit ist es in dieser Konfiguration möglich, eine verschlüsselte Kommunikation auch mit einem Terminal durchzuführen, das kein Programm für eine Verschlüsselungsverarbeitung hat. Zu dieser Zeit werden die Verschlüsselungs- und Entschlüsselungsverarbeitung von dem Agenten ausgeführt, daher ist es nicht notwendig, dass sich der Benutzer mit dem für diese verschlüsselte Kommunikation verwendeten Verschlüsselungsverfahren beschäftigt. Außerdem werden die Verschlüsselungs- und Entschlüsselungsverarbeitung von Agenten mit denselben Funktionen sowohl im Sendeterminal als auch im Empfangsterminal vorgenommen, so dass der chiffrierte Text zuverlässig im Empfangsterminal entschlüsselt werden kann. Das Verschlüsselungsverfahren kann, wenn gewünscht, geheim sein. Die Sicherheit der Verschlüsselung kann durch die Änderung des für die Verschlüsselung benötigten Schlüssels synchron in Übereinstimmung mit im Voraus zwischen den Agenten vereinbarten Regeln erhöht werden. Daher kann ein Verschlüsselungsverfahren mit einem geringen Gesamtaufwand ausgewählt werden, um die Verarbeitungszeit zu reduzieren.

[0037] Im Nachstehenden werden Ausführungsformen dieser Erfindung mit Bezugnahme auf die Zeichnungen beschrieben. **Fig.** 1 ist eine Darstellung, die eine Übersicht über diese Erfindung erläutert. Diese Darstellung zeigt ein Beispiel, bei dem Informationen in verschlüsselter Form zwischen einem Server **10** und einem Client **15** transferiert werden. Der Server **10** und der Client **15** sind beide Computer.

[0038] Der vertrauenswürdige Agent **11** hat ein Programm zum Zweck der Verschlüsselung von Daten und ist in mobilem Code beschrieben. Die Sendeeinheit **12** des vertrauenswürdigen Agenten sendet den vertrauenswürdigen Agenten **11** an den Client **15**. Das Anwendungsprogramm **13** nimmt eine Verarbeitung vor, die das Senden von Daten an den Client **15** und das Empfangen von Daten von diesem begleitet. Anwendungen, die in dieser Ausführungsform vorgesehen sind, enthalten Telefon, Fernsehkonferenzen, Videotransmission, etc., von denen alle eine Echtzeitverarbeitung erfordern, die möglichen Anwendungen sind jedoch nicht auf diese beschränkt. Der vertrauenswürdige Agent **14** ist ein verschlüsseltes Programm, das dieselbe Funktion hat wie der vertrauenswürdige Agent **11**; es ist in dem Server **10** permanent resident.

[0039] Das Anwendungsprogramm **16** ist grundsätzlich gleich wie die Anwendung **13**. Der vertrauenswürdige Agent **17** ist der vertrauenswürdige Agent **11**, der von dem Server **10** transferiert wurde. Der verschlüsselte Kommunikationsweg **18** ist ein Weg, der zwischen dem vertrauenswürdigen Agenten **14** und dem vertrauenswürdigen Agenten **17** hergestellt wird.

[0040] **Fig.** 2 ist eine Konfigurationsdarstellung des Servers **10** und des Clients **15**. Die Speicheranordnung **501** besteht aus einem Halbleiterspeicher, einem magnetischen Aufzeichnungsmedium oder einem optischen Aufzeichnungsmedium, usw., und speichert Programme, Daten, etc. Die Speicheranordnung **501** kann permanent in dem Server **10** oder dem Client **15** installiert sein, oder sie kann entfernbar sein.

[0041] Der Speichermediumtreiber **502** ist eine Anordnung, welche Daten ausliest, die in dem tragbaren Speichermedium **503** (einschließlich Halbleiterspeicher, Magnetplatte, optische Platte, magnetooptische Platte, etc.) gespeichert sind, oder Daten in das tragbare Speichermedium **503** schreibt. Die Kommunikationssteuereinheit **504** ist eine Einheit, die das Senden von Daten an ein Netz und das Empfangen von Daten von diesem steuert.

[0042] Die CPU **505** lädt Programme aus der Speicheranordnung **501** oder dem tragbaren Speichermedium **503** in den Speicher **506** und führt diese aus. Es ist zu beachten, dass Programme und Daten, die in dem Speichermedium **501** gespeichert sind, aus dem tragbaren Speichermedium **503** eingeschrieben worden sein können, oder von einem anderen Gerät an einem Netz über eine Kommunikationsleitung empfangen werden können. Die Konfiguration kann auch so sein, dass die CPU **505** Programme und Daten, die in einer anderen Speicheranordnung gespeichert sind, über eine Kommunikationsleitung verwenden kann.

[0043] Die vertrauenswürdigen Agenten **11** und **14** werden vorher in der Speicheranordnung **501** in dem Server **10** gespeichert. Die vertrauenswürdigen Agenten **11** und **14** können aus dem tragbaren Speichermedium **503** in die Speicheranordnung **501** installiert werden, oder können aus einer anderen Anordnung an einem Netz in die Speicheranordnung **501** installiert werden. Der vertrauenswürdige Agent **14** wird in den Speicher

506 geladen, wenn eine verschlüsselte Kommunikation gestartet wird.

[0044] In dem Client **15** wird der vertrauenswürdige Agent **17** über die Kommunikationssteuersektion **504** empfangen und in den Speicher **506** geladen.

[0045] Die Aktion des in **Fig.** 1 gezeigten Kryptosystems ist wie folgt. Zuerst, vor der Datenkommunikation, wird die Sendeeinheit **12** des vertrauenswürdigen Agenten hochgefahren, und der vertrauenswürdige Agent **11** wird von dem Server **10** an den Client **15** gesendet. Zu dieser Zeit wird der vertrauenswürdige Agent **11** durch ein Verfahren wie RSA oder RSA + DES verschlüsselt und transferiert.

[0046] Die Verarbeitungsgeschwindigkeiten von RSA, DES, etc., sind langsam, daher sind sie nicht die besten Verschlüsselungsverfahren, welche für die Verschlüsselung von Daten zu verwenden sind, die eine Echtzeitverarbeitung erfordern, wie Audiodaten und Videodaten, wenn jedoch ein vertrauenswürdiger Agent verschlüsselt wird, müssen die Verschlüsselungsverarbeitung und die Entschlüsselungsverarbeitung jeweils nur einmal durchgeführt werden, und die Datenmenge ist viel geringer als in den Fällen von Audiodaten und Videodaten, so dass sogar in den Fällen von RSA und DES die Verarbeitungsgeschwindigkeit kein Problem wird.

[0047] Als Nächstes stellen der vertrauenswürdige Agent **14** des Servers **10** und der vertrauenswürdige Agent **17** des Clients **15** den verschlüsselten Kommunikationsweg **18** her. Die Verarbeitung, durch die der vertrauenswürdige Agent **11** von dem Server **10** zum Client **15** transferiert wird, und die Verarbeitung, durch die der verschlüsselte Kommunikationsweg **18** zwischen dem vertrauenswürdigen Agenten **14** und dem vertrauenswürdigen Agenten **17** hergestellt wird, werden im Nachstehenden erläutert.

[0048] Die vertrauenswürdigen Agenten **14** und **17** sind mit den Anwendungsprogrammen **13** bzw. **16** verknüpft; sie verschlüsseln Daten so, dass die Daten nicht gestohlen oder verfälscht werden können, dann senden und empfangen sie die Daten an/voneinander. Die Verschlüsselung zwischen dem vertrauenswürdigen Agenten **14** und dem vertrauenswürdigen Agent **17** folgt dem Verfahren, das in den in den vertrauenswürdigen Agenten **14** und **17** enthaltenen Programmen beschrieben ist. Die vertrauenswürdigen Agenten **14** und **17** können synchron den für die Verschlüsselung notwendigen Schlüssel (einen geheimen Schlüssel) in Übereinstimmung mit einer vorherbestimmten Regel ändern. Dies erhöht die Stärke der Verschlüsselung.

[0049] **Fig.** 3 ist eine Konfigurationsdarstellung des vertrauenswürdigen Agenten **11**, der in mobilem Code beschrieben ist. Der vertrauenswürdige Agent **11**, wie in **Fig.** 3A gezeigt, besteht aus einer Anwendungsschnittstellensektion **11-1** und einer kryptographischen Verarbeitungssektion **11-2**. Die Anwendungsschnittstellensektion **11-1** hat die Rolle des Austauschs von Signalen zwischen einem gewöhnlichen Anwendungsprogramm (hier Anwendung **16**) und der kryptographischen Verarbeitungssektion **11-2**; die kryptographische Verarbeitungssektion **11-2** verschlüsselt und entschlüsselt Signale an die/von der Anwendungsschnittstellensektion **11-1**. Eine weitere Funktion, die die Anwendungsschnittstellensektion **11-1** hat, ist die Absorption von Differenzen aufgrund verschiedener Betriebssysteme, wenn die API von dem Betriebssystem abhängig ist.

[0050] In **Fig.** 3B hat der vertrauenswürdige Agent **11** eine Anwendungsprogrammsektion **11-3**. In diesem Fall ist das in dem vertrauenswürdigen Agenten **11** gespeicherte Anwendungsprogramm in mobilem Code beschrieben, und die Anwendungsprogrammsektion **11-3** wird zusammen mit der Anwendungsschnittstellensektion **11-1** und der kryptographischen-Verarbeitungssektion **11-2** gesendet.

[0051] **Fig.** 4A ist ein Sequenzdiagramm, das die Verarbeitung erläutert, durch die ein vertrauenswürdiger Agent gesendet wird und ein verschlüsselter Kommunikationsweg hergestellt wird. Hier wird angenommen, dass der Server **10** das Terminal ist, das den vertrauenswürdigen Agenten sendet, und der Client **15** ist das Terminal, das den vertrauenswürdigen Agenten empfängt. Ferner wird angenommen, dass das Verschlüsselungsverfahren Pseudozufallszahlen verwendet. Die Pseudozufallszahlen werden auf der Basis eines "Initialstartparameters" generiert, wie detailliert im Nachstehenden erläutert wird.

[0052] Wenn die verschlüsselte Kommunikation gestartet wird, wird zuerst ein Initialstartparameter in dem Server **10** generiert. Der Initialstartparameter wird beispielsweise auf der Basis der Zeit generiert. Als Nächstes wird dieser Initialstartparameter in den vertrauenswürdigen Agenten **11** und **14** eingestellt. Dann wird der vertrauenswürdige Agent **11**, in dem der Initialstartparameter eingestellt wurde, an den Client **15** gesendet. Der vertrauenswürdige Agent **11** wird, wie oben diskutiert, transferiert, nachdem er durch das RSA- oder DES-Verfahren verschlüsselt wurde. Dann startet der Server **10** den vertrauenswürdigen Agenten **14**.

[0053] Es wird angenommen, dass in dem Client **15** ein Programm zum Empfangen des Agenten hochgefah-

ren wurde. Wenn der Client **15** den vertrauenswürdigen Agenten **11** empfängt, wird dieser vertrauenswürdige Agent **11** in den Speicher geladen und als vertrauenswürdiger Agent **17** hochgefahren.

[0054] Die vertrauenswürdigen Agenten **14** und **17** stellen einen verschlüsselten Kommunikationsweg her; danach wird chiffrierter Text über diesen verschlüsselten Kommunikationsweg gesendet und empfangen.

[0055] **Fig.** 4B ist eine Darstellung, die den Vorgang erläutert, durch den ein verschlüsselter Kommunikationsweg zwischen vertrauenswürdigen Agenten hergestellt wird. Hier wird angenommen, dass der vertrauenswürdige Agent **14** bereits hochgefahren wurde. Die kryptographischen Verarbeitungssektionen der vertrauenswürdigen Agenten **14** und **17** bestehen jeweils aus einer Sendesektion und einer Empfangssektion. Die Sendesektion verschlüsselt Daten von einem Anwendungsprogramm und nimmt eine Sendeverarbeitung vor; die Empfangssektion konvertiert verschlüsselte Daten in Klartext oder geeignete Anwendungsdaten und nimmt eine Verarbeitung vor, um die Daten zu einem Anwendungsprogramm zu transferieren. Die Sendesektion und die Empfangssektion werden beispielsweise durch Threads realisiert.

[0056] Zuerst sendet die Sendesektion des vertrauenswürdigen Agenten **14** eine Verbindungsanforderung an die Empfangssektion des vertrauenswürdigen Agenten **17**. Wenn der Transmissionsweg beispielsweise ein Ethernet ist, wird diese Verbindungsanforderung durch ein TCP-Paket transferiert. Da zu dieser Zeit die Verbindungsanforderung abgelehnt wird, wenn der vertrauenswürdige Agent **17** nicht hochgefahren wurde, erteilt in diesem Fall die Sendesektion des vertrauenswürdigen Agenten **14** wiederholt die Verbindungsanforderung, bis eine Antwort von der Empfangssektion des vertrauenswürdigen Agenten **17** empfangen wird.

[0057] Wenn die Empfangssektion des vertrauenswürdigen Agenten **17** eine Antwortnachricht ansprechend auf die Verbindungsanforderung sendet, und die Empfangssektion des vertrauenswürdigen Agenten **14** diese Nachricht empfängt, wird ein Weg zwischen der Sendesektion des vertrauenswürdigen Agenten **14** und der Empfangssektion des vertrauenswürdigen Agenten **17** hergestellt. Der Vorgang zur Herstellung eines Wegs zwischen der Sendesektion des vertrauenswürdigen Agenten **17** und der Empfangssektion des vertrauenswürdigen Agenten **14** ist ähnlich.

[0058] Danach verschlüsselt die Sendesektion des vertrauenswürdigen Agenten **14** Daten von dem Anwendungsprogramm **13** und sendet dann die verschlüsselten Daten an den vertrauenswürdigen Agenten **17** über den Weg, der hergestellt wurde. Die Empfangssektion des vertrauenswürdigen Agenten **17** entschlüsselt den empfangenen chiffrierten Text oder verschlüsselte Daten und transferiert den erhaltenen Klartext oder geeignete Anwendungsdaten zum Anwendungsprogramm **16**. Der Vorgang zum Senden verschlüsselter Daten in der umgekehrten Richtung ist ähnlich. Die verschlüsselten Daten werden beispielsweise in einem UDP-Paket gespeichert und dann transferiert.

[0059] **Fig.** 5 ist eine Konfigurationsdarstellung des verschlüsselten Kommunikationssystems einer Ausführungsform dieser Erfindung. Die Arbeitsstationen **20** und **30** entsprechen dem Server **10** bzw. dem Client **15** in **Fig.** 1. Die Anwendungsprogramme **21** und **31** entsprechen den Anwendungsprogrammen **13** bzw. **16** in **Fig.** 1. Die vertrauenswürdigen Agenten **22** und **32** entsprechen den vertrauenswürdigen Agenten **14** bzw. **17** in **Fig.** 1. Die Arbeitsstation **20** und die Arbeitsstation **30** sind über das Internet **19** miteinander verbunden.

[0060] Der vertrauenswürdige Agent **22** hat die kryptographischen Verarbeitungseinheiten **26** bis **29**. Jede der kryptographischen Verarbeitungseinheiten **26** bis **29** verschlüsselt Daten durch ein von den anderen verschiedenes Verfahren. Die Verschlüsselungsverfahrensauswahl-Einheit **24** wählt eine der kryptographischen Verarbeitungseinheiten **26** bis **29** in Übereinstimmung mit einer Instruktion von der Verschlüsselungsverfahrensauswahl-Steuereinheit **25** aus, und transferiert über die Anwendungsschnittstellensektion **23** empfangene Daten zur ausgewählten kryptographischen Verarbeitungseinheit. Die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** generiert ein Instruktionssignal und gibt dieses aus zum Zweck der Auswahl einer unter den kryptographischen Verarbeitungseinheiten **26** bis **29** in Übereinstimmung mit einem spezifizierten Algorithmus. Das Verfahren, durch das dieses Instruktionssignal generiert wird, wird im Nachstehenden beschrieben. Die Anwendungsschnittstellensektion **23** ist grundsätzlich gleich wie die in **Fig.** 3A und **Fig.** 3B gezeigte Anwendungsschnittstellensektion **11-1**.

[0061] Der vertrauenswürdige Agent **32** wurde von der Arbeitsstation **20** transferiert. Die Anwendungsschnittstelleneinheit **33**, die Verschlüsselungsverfahrensauswahl-Einheit **34** und die kryptographischen Verarbeitungseinheiten **35** bis **38** sind grundsätzlich jeweils gleich wie die Anwendungsschnittstelleneinheit **23**, die Verschlüsselungsverfahrensauswahl-Einheit **24** und die kryptographischen Verarbeitungseinheiten **26** bis **29**, die den vertrauenswürdigen Agenten **22** bilden. Der vertrauenswürdige Agent **32** hat keine Einheit, die der Ver-

schlüsselungsverfahrensauswahl-Steuereinheit **25** entspricht; die Verschlüsselungsverfahrensauswahl-Einheit **34** wählt eine der kryptographischen Verarbeitungseinheiten **35** bis **38** in Übereinstimmung mit einem Instruktionssignal aus, das von der Verschlüsselungsverfahrensauswahl-Steuereinheit **25** generiert wird.

[0062] Die Aktion des in **Fig.** 5 gezeigten Verschlüsselungskommunikationssystems ist wie folgt. Zuerst wird der vertrauenswürdige Agent **32**, der in mobilem Code beschrieben ist, verschlüsselt und von der Arbeitsstation **20** zur Arbeitsstation **30** transferiert.

[0063] Als Nächstes wird das Verschlüsselungsverfahren, das für die verschlüsselte Kommunikation notwendig ist, in der Verschlüsselungsverfahrensauswahl-Steuereinheit **25** bestimmt. Die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** transferiert Informationen, die anweisen, welches Verschlüsselungsverfahren zu verwenden ist, zur Verschlüsselungsverfahrensauswahl-Einheit **24** und zur Verschlüsselungsverfahrensauswahl-Einheit **34** der Arbeitsstation **30**. Dies sichert den verschlüsselten Kommunikationsweg.

[0064] Wenn die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** beispielsweise das erste Verschlüsselungsverfahren auswählt, dann werden, wie in **Fig.** 5 gezeigt, in der Arbeitsstation **20** Daten unter Verwendung der 1. kryptographischen Verarbeitungseinheit **26** verschlüsselt, und in der Arbeitsstation **30** werden der chiffrierte Text oder verschlüsselte Daten unter Verwendung der 1. kryptographischen Verarbeitungseinheit **35** entschlüsselt.

[0065] Kommunikationsdaten von dem Anwendungsprogramm **21** werden in der kryptographischen Verarbeitungseinheit verschlüsselt, die von der Verschlüsselungsverfahrensauswahl-Einheit **24** über die Anwendungsschnittstelleneinheit **23** ausgewählt wird. Das in **Fig.** 5 gezeigte Beispiel zeigt den Fall, in dem die 1. kryptographische Verarbeitungseinheit **26** ausgewählt wurde.

[0066] Die verschlüsselten Daten werden über das Internet **19** an die Arbeitsstation **30** gesendet. Daten (chiffrierter Text oder verschlüsselte Daten), die von der Arbeitsstation **30** empfangen werden, werden in der kryptographischen Verarbeitungseinheit entschlüsselt, die von der Verschlüsselungsverfahrensauswahl-Einheit **34** ausgewählt wird (in diesem Beispiel die 1. kryptographische Verarbeitungseinheit **35**), und zum Anwendungsprogramm **31** in der Arbeitsstation **30** über die Anwendungsschnittstelleneinheit **33** transferiert.

[0067] Die Datentransmission von dem Anwendungsprogramm **31** in der Arbeitsstation **30** zum Anwendungsprogramm **21** in der Arbeitsstation **20** wird durch einen ähnlichen Verschlüsselungsvorgang durchgeführt.

[0068] Die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** wählt neue Verschlüsselungsverfahren in regelmäßigen oder unregelmäßigen Intervallen aus, und übermittelt der Verschlüsselungsverfahrensauswahl-Einheit **24** der Arbeitsstation **20** und der Verschlüsselungsverfahrensauswahl-Einheit **34** der Arbeitsstation **30** die Auswahlergebnisse. Bei dieser Konfigurationsart ändert sich das Verschlüsselungsverfahren mit der Zeit, wodurch es schwierig wird, die verschlüsselten Daten zu entschlüsseln.

[0069] In der oben beschriebenen Ausführungsform ist die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** in der Arbeitsstation **20** vorgesehen, es ist jedoch beispielsweise auch möglich, einen Verschlüsselungsverfahrensinstruktions-Server am Netz zu installieren, und zu bewirken, dass die vertrauenswürdigen Agenten ihre Verschlüsselungsverfahren auf der Basis von Instruktionen von dem Verschlüsselungsverfahrensinstruktions-Server umschalten.

[0070] Auch sind in der oben beschriebenen Ausführungsform die kryptographischen Verarbeitungseinheiten innerhalb der vertrauenswürdigen Agenten, es ist jedoch auch möglich, einen Verteilungsserver für die kryptographischen Verarbeitungseinheiten zu installieren, der Programme zum Zweck der kryptographischen Verarbeitung, die in mobilem Code beschrieben sind, am Netz verteilt, und Programme zum Zweck der kryptographischen Verarbeitung zu haben, die von dem Verteilungsserver für die kryptographischen Verarbeitungseinheiten an die vertrauenswürdigen Agenten verteilt werden.

[0071] **Fig.** 6 ist eine Konfigurationsdarstellung des verschlüsselten Kommunikationssystems einer weiteren Ausführungsform dieser Erfindung. In **Fig.** 6 gelten die vorhergehenden Erläuterungen ohne Änderung für Komponenten mit denselben Symbolen, die in **Fig.** 5 verwendet wurden.

[0072] In dem in **Fig.** 6 gezeigten System ist die Verschlüsselungsverfahrensauswahl-Steuereinheit **39** in dem vertrauenswürdigen Agenten **32** vorgesehen. Die Verschlüsselungsverfahrensauswahl-Steuereinheit **39** ist gleich wie die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** innerhalb des vertrauenswürdigen

Agenten **22**. Demgemäß kann der vertrauenswürdige Agent **32** dasselbe Verschlüsselungsverfahren selbst auswählen wie das vom vertrauenswürdigen Agenten **22** ausgewählte Verschlüsselungsverfahren, ohne eine Instruktion zum Zweck der Verschlüsselungsverfahrensauswahl von dem vertrauenswürdigen Agenten **22** zu empfangen.

[0073] Nun wird die Aktion des in **Fig.** 6 gezeigten verschlüsselten Kommunikationssystems erläutert. Zuerst wird der vertrauenswürdige Agent **32** von der Arbeitsstation **20** zur Arbeitsstation **30** transferiert. Diese Verarbeitung ist wie mit Bezugnahme auf **Fig.** 5 erläutert.

[0074] Die Verschlüsselungsverfahrensauswahl-Steuereinheit **25** in der Arbeitsstation **20** und die Verschlüsselungsverfahrensauswahl-Steuereinheit **39** in der Arbeitsstation **30** bestimmen ihre jeweiligen Verschlüsselungsverfahren unabhängig voneinander, und übermitteln der Verschlüsselungsverfahrensauswahl-Einheit **24** der Arbeitsstation **20** bzw. der Verschlüsselungsverfahrensauswahl-Einheit **34** der Arbeitsstation **30** diese jeweiligen Verschlüsselungsverfahren, die bestimmt wurden. Hier haben die Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** Synchronisationsfunktion bei der Verschlüsselungsverfahrensauswahl, so dass dasselbe Verschlüsselungsverfahren von den Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **34** ausgewählt wird.

[0075] Nun wird einfach erläutert, was mit "Synchronisationsfunktionen bei der Verschlüsselungsverfahrensauswahl" gemeint ist. Die Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** geben jeweils Ergebnisse aus, die in Übereinstimmung mit gegebenen Initialbedingungen erhalten wurden. Die der Verschlüsselungsverfahrensauswahl-Steuereinheit **39** gegebenen Initialbedingungen werden in der Arbeitsstation **20** eingestellt. Diese Initialbedingungen sind gleich wie jene, die der Verschlüsselungsverfahrensauswahl-Steuereinheit **25** gegeben werden. In **Fig.** 4, die oben diskutiert wurde, wurden dieselben Initialstartparameter in 2 vertrauenswürdigen Agenten als Initialbedingungen eingestellt. Hier haben die Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** einander gleiche Funktionen, so dass, wenn den Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** dieselben Initialbedingungen gegeben werden, sie dieselben Ergebnisse generieren. Demgemäß arbeiten die Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** unabhängig voneinander, geben jedoch dieselben Werte als Signale aus, die das zu verwendende Verschlüsselungsverfahren anzeigen. Dies wird als Synchronisation bei der Auswahl des Verschlüsselungsverfahrens bezeichnet.

[0076] Mittels der Synchronisationsfunktion wird immer dasselbe Verschlüsselungsverfahren in der Arbeitsstation **20** und der Arbeitsstation **30** ausgewählt, ohne dass Informationen zwischen ihnen gesendet und empfangen werden. Dies sichert einen verschlüsselten Kommunikationsweg.

[0077] Die Aktion, durch die Daten verschlüsselt und zwischen den Anwendungsprogrammen **21** und **31** gesendet und empfangen werden, ist wie in **Fig.** 5 erläutert. Das heißt, Kommunikationsdaten von dem Anwendungsprogramm **21** werden in der kryptographischen Verarbeitungseinheit verschlüsselt, die von der Verschlüsselungsverfahrensauswahl-Einheit **24** ausgewählt wird. Die verschlüsselten Daten werden über das Internet **19** zur Arbeitsstation **30** gesendet. Die Daten (chiffrierter Text oder chiffrierte Daten), die von der Arbeitsstation **30** empfangen werden, werden in der kryptographischen Verarbeitungseinheit entschlüsselt, die von der Verschlüsselungsverfahrensauswahl-Einheit **34** ausgewählt wird, und zum Anwendungsprogramm **31** in der Arbeitsstation **30** transferiert.

[0078] Somit unterscheidet sich das in **Fig.** 6 gezeigte System von dem in **Fig.** 5 gezeigten System dadurch, dass die Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** voneinander unabhängig sind, und die Verschlüsselungsverfahren sequentiell ausgewählt werden. Die Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39** wählen in regel mäßigen oder unregelmäßigen Intervallen neue Verschlüsselungsverfahren aus, und senden diese Auswahlergebnisse an die Verschlüsselungsverfahrensauswahl-Einheit **24** bzw. die Verschlüsselungsverfahrensauswahl-Einheit **34**. In dieser Konfiguration ändert sich das Verschlüsselungsverfahren mit der Zeit, wodurch es schwierig wird, die verschlüsselten Daten zu entschlüsseln.

[0079] **Fig.** 7 zeigt ein Konfigurationsbeispiel einer kryptographischen Verarbeitungseinheit. Die folgende Diskussion nimmt an, dass Pseudozufallszahlen in dem Verschlüsselungsverfahren verwendet werden.

[0080] Theoretisch, wie beispielsweise in **Fig.** 7A gezeigt, besteht die kryptographische Verarbeitungseinheit aus einem Exklusiv-ODER-Generator **40** und einem Pseudozufallszahlengenerator **41**. Der Pseudozufallszahlengenerator **41** kann ein variabler Periodentyp sein. Die verschlüsselten Daten (chiffrierter Text) werden durch das Eingeben der zu verschlüsselnden Daten (Klartext) und Pseudozufallszahlen, die von dem Pseudozufalls-

zahlengenerator **41** generiert werden, in den Exklusiv-ODER-Generator **40** erhalten. Die Konfiguration ist grundsätzlich gleich, wenn der chiffrierte Text in Klartext entschlüsselt wird.

[0081] **Fig.** 7B zeigt ein weiteres Beispiel einer kryptographischen Verarbeitungseinheit. In diesem Beispiel hat die kryptographische Verarbeitungseinheit, zusätzlich zu dem Exklusiv-ODER-Generator **40** und dem Pseudozufallszahlengenerator **41**, eine Startparametersektion **42**, die Startparameter zum Zweck des Generierens von Pseudozufallszahlen generiert, und eine Startparameteränderungssektion **43**, die Instruktionen ausgibt, um die Startparameter, die in der Startparametersektion **42** generiert werden, in unregelmäßigen Intervallen zu ändern.

[0082] In der in **Fig.** 7B gezeigten kryptographischen Verarbeitungseinheit kann die Periode der Pseudozufallszahlen geändert werden, indem die Startparametersektion **42** und die Startparameteränderungssektion **43** vorgesehen werden, wodurch es schwierig wird, die verschlüsselten Daten zu entschlüsseln. Die Aktion, durch die Klartext unter Verwendung des Exklusiv-ODER-Generators **40** und des Pseudozufallszahlengenerators **41** verschlüsselt wird, ist gleich wie in dem in **Fig.** 7A gezeigten Fall.

[0083] Nun wird das Verfahren zur Änderung der Periode der Pseudozufallszahlen erläutert. In einem Fall, in dem der Pseudozufallszahlengenerator durch eine Hardware-Schaltung realisiert ist, wird die Periode des Pseudozufallszahlengenerators beispielsweise durch die Änderung der Anzahl von Stufen und der Verdrahtung in dem linearen rückgekoppelten Schieberegistersystem bestimmt, das die Pseudozufallszahlen generiert.

[0084] Ein Beispiel davon ist in **Fig.** 8 gezeigt. **Fig.** 8 zeigt einen Fall, in dem eine Ausführungsform eines Pseudozufallszahlengenerators durch eine Hardware-Schaltung realisiert ist. In **Fig.** 8 ist **44** ein Schieberegister, **45** ist eine Wegsteuersektion, s0 bis s12 und sa bis sl sind Schalter zur Wegverbindung, x1 bis x12 sind Exklusiv-ODER-Schaltungen, und r1 bis r13 sind Bitelemente des Schieberegisters **44**. In der in **Fig.** 8 gezeigten Schaltung wird das von der Wegsteuersektion **45** generierte Signal verwendet, und die Periode der Pseudozufallszahlen wird durch die Steuerung der Rückkopplung von r1 bis r13 in dem Schieberegister **44** geändert, indem die Verbindung und Trennung der Wege mittels der Schalter s0 bis s12 und der Schalter sa bis sl gesteuert werden.

[0085] Nun wird die Realisierung eines Pseudozufallszahlengenerators behandelt, der eine 3-stufige M-Serie generiert. Ein primitives Polynom, das eine 3-stufige M-Serie generiert, ist $x^3 + x + 1$; die Hardware-Konfiguration ist wie in **Fig.** 9 gezeigt. In **Fig.** 9 ist **44** ein Schieberegister, und **46** ist eine Exklusiv-ODER-Schaltung. Demgemäß werden in dem in **Fig.** 8 gezeigten Pseudozufallszahlengenerator, um die in **Fig.** 9 gezeigte Konfiguration zu realisieren, der Schalter s2 und der Schalter sb EIN geschaltet, und die anderen Schalter werden AUS geschaltet.

Primitive Polynome, die eine n-stufige M-Serie generieren, und ihre Perioden werden nachstehenden angegeben

[Anzahl von Stufen n]	[primitives Polynom]	[Periode]
2	$x^2 + x + 1$	3
3	$x^3 + x + 1$	7
4	$x^4 + x + 1$	15
5	$x^5 + x^2 + 1$	31
6	$x^6 + x + 1$	63
7	$x^7 + x^3 + 1$	127
8	$x^8 + x^4 + x^3 + x^2 + 1$	255
9	$x^9 + x^4 + 1$	511
10	$x^{10} + x^3 + 1$	1023
11	$x^{11} + x^2 + 1$	2047
12	$x^{12} + x^6 + x^4 + x + 1$	4055

[0086] Um beispielsweise einen 6-stufigen Pseudozufallszahlengenerator zu realisieren, ist es ausreichend, die in **Fig.** 8 gezeigte Summe von r6 und r1 zu r6 zurückzuführen, somit ist es ausreichend, den Schalter s5 und den Schalter se EIN zu schalten.

[0087] In **Fig.** 10 sind einige Beispiele von Pseudozufallszahlengeneratoren gezeigt, die als Kombinationen von Pseudozufallszahlengeneratoren erhalten werden.

[0088] In der in **Fig.** 10A gezeigten Konfiguration wird der Ausgang des in **Fig.** 8 gezeigten Pseudozufallszahlengenerators wie er ist als Pseudozufallszahlen verwendet. In der in **Fig.** 10B gezeigten Konfiguration werden die 2 Ausgänge des Pseudozufallszahlengenerators **41a** und des Pseudozufallszahlengenerators **41b** in die Exklusiv-ODER-Schaltung **47** eingegeben, und der Ausgang dieser Exklusiv-ODER-Schaltung **47** wird als Pseudozufallszahlen verwendet. In diesem Fall wird beispielsweise davon ausgegangen, dass die in dem Pseudozufallszahlengenerator **41a** und dem Pseudozufallszahlengenerator **41b** eingestellten Initialstartparameter voneinander verschieden sind. In der in **Fig.** 10C gezeigten Konfiguration werden die 3 Pseudozufallszahlengeneratoren **41c**, **41d** und **41e** und der Schalter **48** verwendet; die Ausgänge aus den 2 Pseudozufallszahlengeneratoren **41c** und **41d** werden in den Schalter **48** eingegeben. Der Ausgang des Pseudozufallszahlengenerators **41e** wird verwendet, um den Schalter **48** zu steuern und den Ausgang entweder des Pseudozufallszahlengenerators **41c** oder des Pseudozufallszahlengenerators **41d** auszuwählen. Dann wird der Ausgang des Schalters **48** als Pseudozufallszahlen verwendet.

[0089] In dem vertrauenswürdigen Agenten dieser Ausführungsform gibt es ein Software-Programm, um die oben beschriebene Aktion zu realisieren, und Pseudozufallszahlen werden durch das Ausführen dieses Programms generiert.

[0090] Die in **Fig.** 5 und **Fig.** 6 gezeigten vertrauenswürdigen Agenten haben eine Vielzahl kryptographischer Verarbeitungseinheiten; es ist beispielsweise möglich, die in **Fig.** 10A bis 10C gezeigten Zufallszahlen-Generiersysteme als Pseudozufallszahlen-Generierquellen in der ersten, zweiten bzw. dritten kryptographischen Verarbeitungseinheit zu verwenden.

[0091] **Fig.** 11 ist ein Flussdiagramm, das die Aktion einer Verschlüsselungsverfahrensauswahl-Steuereinheit zeigt. Hier wird die Aktion der Verschlüsselungsverfahrensauswahl-Steuereinheit **25** in **Fig.** 5 erläutert.

[0092] In Schritt S1 wird ein Initialstartparameter auf der Basis der Zeit, des Datums, des Wochentags, etc., geschaffen, die durch die interne Uhr in der Arbeitsstation **20** angezeigt werden. Dieser Initialstartparameter wird innerhalb der Verschlüsselungsverfahrensauswahl-Steuereinheit **25** eingestellt. In Schritt S2 wird ein

Pseudozufallszahlengenerator verwendet, um Pseudozufallszahlen aus dem in Schritt S1 geschaffenen Initialstartparameter zu generieren. In Schritt S3 wird das Verschlüsselungsverfahren auf der Basis der in Schritt S2 generierten Pseudozufallszahlen ausgewählt. In Schritt S4 werden Informationen, die das in Schritt S3 ausgewählte Verschlüsselungsverfahren identifizieren, zu den Verschlüsselungsverfahrensauswahl-Steuereinheiten **24** und **34** transferiert.

[0093] In Schritt S5 wird die Zeiteinstellung bestimmt, zu der das Verschlüsselungsverfahren umgeschaltet wird. Diese Schaltzeiteinstellung wird im Nachstehenden detaillierter erläutert; sie wird hinsichtlich eines Parameters wie der Anzahl von Paketen oder der Zeit ausgedrückt. In Schritt S6 wird überwacht, ob die kryptographische Verarbeitungssequenz die Zeit zur Änderung des Verschlüsselungsverfahrens erreicht hat oder nicht. Wenn die kryptographische Verarbeitungssequenz die Zeit zum Umschalten des Verschlüsselungsverfahrens erreicht, wird die Pseudozufallszahl, die unmittelbar vor dieser Zeit generiert wurde, als Startparameter in Schritt S7 eingestellt, und dann kehrt die Prozedur zu Schritt S2 zurück. Danach werden die Schritte S2 bis S7 wiederholt.

[0094] Mittels der oben beschriebenen Verarbeitung wird das Verschlüsselungsverfahren gemäß der Pseudozufallszahl ausgewählt; das Verschlüsselungsverfahren wird dann wiederholt gemäß der durch die Pseudozufallszahlen bestimmten Zeiteinstellung umgeschaltet.

[0095] Das Verfahren zur Auswahl des Verschlüsselungsverfahrens in Schritt S3 und das Verfahren zur Bestimmung der Schaltzeiteinstellung in Schritt S5 sind beispielsweise wie folgt. Wenn Pseudozufallszahlen des 64 Bit langen Typs verwendet werden, ist der Bereich der Werte, die durch die Abtastung der Pseudozufallszahlen erhalten werden, $-9{,}223372035 \times 10^{18}$ bis $+9{,}223372035 \times 10^{18}$. Demgemäß werden beispielsweise in einem Fall, in dem 10 Verschlüsselungsverfahren bestehen, Werte von 1 bis 10 aus den Pseudozufallszahlen erhalten, indem herangezogen wird:
ausgewählte Zahl = random/10^{18} + 1

[0096] Hier ist "random" eine durch den Pseudozufallszahlengenerator generierte Pseudozufallszahl. Wenn es 5 Verschlüsselungsverfahren gibt, dann werden Werte von 1 bis 5 aus den Pseudozufallszahlen erhalten, indem herangezogen wird:
ausgewählte Zahl = (random/10^{18})/2 + 1

[0097] Wenn beispielsweise die Anzahl von Paketen zur Bestimmung der Schaltzeiteinstellung verwendet wird, dann wird, in einer Prozedur ähnlich der zur Auswahl der Nummer des Verschlüsselungsverfahrens verwendeten, herangezogen:
Anzahl von Paketen = random/10^{17} + 1

[0098] Dies ergibt einen Wert von 1 bis 92 auf der Basis der Pseudozufallszahl.

[0099] Beispielsweise wird davon ausgegangen, dass "3" als ausgewählte Nummer des Verschlüsselungsvorgangs in Schritt S3 erhalten wurde, und dass "13" als Anzahl von Paketen in Schritt S5 erhalten wurde. Wenn in diesem Fall auf der Sendeseite 13 Pakete gesendet wurden, in denen Daten gespeichert sind, die unter Verwendung der dritten kryptographischen Verarbeitungseinheit verschlüsselt wurden, wird eine andere kryptographische Verarbeitungseinheitennummer ausgewählt. Wenn dann "2" als ausgewählte Nummer in Schritt **3** erhalten wird, werden unter Verwendung der zweiten kryptographischen Verarbeitungseinheit verschlüsselte Daten bis zur nächsten Schaltzeit ausgegeben. Wenn ähnlich auf der Empfangsseite 13 Pakete, in denen verschlüsselte Daten gespeichert sind, unter Verwendung der dritten kryptographischen Verarbeitungseinheit entschlüsselt wurden, wird eine andere kryptographische Verarbeitungseinheitennummer (hier "2") ausgewählt. Dann werden empfangene Pakete unter Verwendung der zweiten kryptographischen Verarbeitungseinheit bis zur nächsten Schaltzeit entschlüsselt.

[0100] In dem Fall von Pseudozufallszahlen, im Gegensatz zu echten Zufallszahlen, werden, sobald der Initialstartparameter und der Generieralgorithmus bestimmt werden, die Pseudozufallszahlen, die aus diesem Generieralgorithmus erhalten werden, einzigartig bestimmt. In der in **Fig.** 6 gezeigten Konfiguration wird diese Eigenschaft von Pseudozufallszahlen verwendet. Das heißt, da die vertrauenswürdigen Agenten **22** und **32** Pseudozufallszahlengeneratoren mit demselben Algorithmus haben, wie oben beschrieben, wenn dieselben Werte als Initialstartparameter eingestellt werden, wird danach das Verschlüsselungsverfahren mit derselben Zeiteinstellung in den vertrauenswürdigen Agenten **22** und **32** umgeschaltet.

[0101] Als Verfahren zur Einstellung des Initialstartparame ters, nachdem derselbe Wert in den vertrauens-

würdigen Agenten **22** und **32** in der Arbeitsstation **20** eingestellt ist, wird der vertrauenswürdige Agent **32** zur Arbeitsstation **30** transferiert. Oder, alternativ dazu, kann die Konfiguration so sein, dass ein Befehl zum Generieren des Initialstartparameters in die vertrauenswürdigen Agenten **22** und **32** eingeführt wird, und die vertrauenswürdigen Agenten **22** und **32** dann ihre Initialstartparameter unabhängig generieren. Wenn in diesem Fall der Befehl beispielsweise einer ist, der den Initialstartparameter in Übereinstimmung mit "heutiges Datum" und "aktuelle Zeit" generiert, dann werden, solange die Uhren in den Arbeitsstationen **20** und **30** korrekt funktionieren, identische Zufallszahlen in den vertrauenswürdigen Agenten **22** und **32** generiert, und dieselben Verschlüsselungsverfahren werden ausgewählt.

[0102] Als Nächstes wird mit Bezugnahme auf **Fig.** 12 und 13 die Sequenz erläutert, wenn Daten zwischen Anwendungsprogrammen gesendet und empfangen werden. Hier wird der Fall behandelt, in dem Daten von dem Anwendungsprogramm **21** in der Arbeitsstation **20** zum Anwendungsprogramm **31** in der Arbeitsstation **30** gesendet werden.

[0103] Die Daten von dem Anwendungsprogramm **21** werden, wie in **Fig.** 12A gezeigt, zum Zweck des Speicherns in Pakete segmentiert. Hier wird als Beispiel angenommen, dass UDP (User Datagram Protocol) als Datentransferprotokoll verwendet wird. Als Nächstes werden Daten durch das spezifizierte Verfahren verschlüsselt, ein Segment zu einem Zeitpunkt. Eine Sequenznummer wird jedem Datensegment zugeordnet. Die Sequenznummern werden so verwendet, dass eine kryptographische Synchronisation zwischen der Sendeseite und der Empfangsseite hergestellt werden kann, auch wenn ein Paket verloren gehen sollte. Das heißt, das UDP-Protokoll ist geeignet, wenn eine Echtzeitverarbeitung erfordernde Daten wie Audiodaten und Videodaten gesendet werden, da es jedoch keine Funktion zum erneuten Senden hat, wenn ein Paket bei der Transmission verloren geht, wird es unmöglich, die Daten auf der Empfangsseite zu reproduzieren. Aus diesem Grund wird eine Sequenznummer jedem Datensegment zugeordnet, so dass die Empfangsseite den Verlust von Paketen detektieren und die Daten korrekt reproduzieren kann.

[0104] Danach wird ein Kopf hinzugefügt und zur Arbeitsstation **30** gesendet. Ein Beispiel einer Paketkonfiguration ist in **Fig.** 12B gezeigt. Die Sequenznummern und der Kopf werden nicht verschlüsselt.

[0105] In der oben beschriebenen Verarbeitung führt der vertrauenswürdige Agent **22** die Verschlüsselungsverarbeitung und die Verarbeitung zur Zuordnung von Sequenznummern durch. Es ist möglich, die Funktionen des vertrauenswürdigen Agenten **22** zu expandieren, so dass die gesamte in **Fig.** 12A gezeigte Verarbeitung von dem vertrauenswürdigen Agenten **22** ausgeführt wird.

[0106] Wenn die Arbeitsstation **30** ein Paket empfängt, wird aus seiner Sequenznummer beurteilt, ob ein Paket bei der Transmission verloren gegangen ist oder nicht. Wenn kein Verlust detektiert wird, wird die verschlüsselte Datensektion extrahiert, und die Daten werden entschlüsselt. Dann werden die entschlüsselten Daten assembliert und zum Anwendungsprogramm **31** transferiert.

[0107] In der oben beschriebenen Verarbeitung nimmt der vertrauenswürdige Agent **32** die Verarbeitung zur Prüfung der Sequenznummern und die Entschlüsselungsverarbeitung vor. Es ist möglich, die Funktionen des vertrauenswürdigen Agenten **32** zu expandieren, so dass der vertrauenswürdige Agent **32** die gesamte in **Fig.** 13 gezeigte Verarbeitung ausführt.

[0108] **Fig.** 14A ist eine Darstellung zur Erläuterung des Verfahrens zur Herstellung einer kryptographischen Synchronisation. Wenn bei der verschlüsselten Kommunikation unter Verwendung von Pseudozufallszahlen Pakete unter Verwendung von Pseudozufallszahlen auf der Sendeseite (der Verschlüsselungsseite) verschlüsselt werden, ist es notwendig, dieselben Pseudozufallszahlen wie die Pseudozufallszahlen zu verwenden, die auf der Sendeseite verwendet werden, wenn diese Pakete auf der Empfangsseite (der Entschlüsselungsseite) entschlüsselt werden. Die vertrauenswürdigen Agenten **22** und **32** generieren dieselben Pseudozufallszahlen in derselben Sequenz mit derselben Zeiteinstellung, und führen eine Verschlüsselungs- und Entschlüsselungsverarbeitung in ihren jeweiligen Sequenzen aus. Dies stellt eine kryptographische Synchronisation her.

[0109] Wenn ein Paket bei der Transmission verloren gegangen ist, dann wird, wie in **Fig.** 14A gezeigt, auf der Entschlüsselungsseite detektiert, welches Paket verloren gegangen ist, und die Entschlüsselungsverarbeitung unter Verwendung der Pseudozufallszahl, die dem Paket entspricht, das verloren gegangen ist, wird übersprungen. In dem in **Fig.** 14A gezeigten Beispiel ist das Paket **3** verloren gegangen; auf der Entschlüsselungsseite wird random(3) nicht verwendet, sondern statt dessen wird eine Entschlüsselungsverarbeitung unter Verwendung von random(4) in Bezug auf das Paket **4** vorgenommen.

[0110] Wenn die Reihenfolge von Paketen während der Transmission vertauscht wird, dann wird, wie in **Fig.** 14B gezeigt, nachdem das Paket **1** entschlüsselt wird, wenn das Paket **3** empfangen wird, wenn das Paket **2** empfangen werden sollte, die Entschlüsselungsverarbeitung unter Verwendung von random(2) übersprungen. Als Nächstes wird das Paket **2** empfangen, wenn normalerweise das Paket **3** empfangen worden wäre, somit wird die Entschlüsselungsverarbeitung unter Verwendung von random(3) übersprungen. Wenn danach das Paket **4** empfangen wird, wenn das Paket **4** empfangen werden sollte, wird ab dieser Zeit die Entschlüsselungsverarbeitung unter Verwendung von random(4) normal durchgeführt.

[0111] Somit wird die Reihenfolge empfangener Pakete auf der Entschlüsselungsseite überwacht; wenn ein Verlust oder Vertauschen der Reihenfolge auftritt, wird die Synchronisation der Verschlüsselungsverarbeitung und Entschlüsselungsverarbeitung durch das Überspringen der Entschlüsselungsverarbeitung aufrechterhalten. Diese Synchronisationsverarbeitung wird auch in dem Fall durchgeführt, wenn das erste Paket verloren geht.

[0112] Um es in dem in **Fig.** 5 oder **Fig.** 6 gezeigten System schwierig zu machen, die verschlüsselten Daten zu entschlüsseln, wurde das Verschlüsselungsverfahren in regelmäßigen oder unregelmäßigen Intervallen geändert. In einem Verschlüsselungsverfahren, bei dem die Verwendung von Pseudozufallszahlen eingesetzt wird, ist es möglich zu bewirken, dass die verschlüsselten Daten schwer zu entschlüsseln sind, indem der zum Generieren dieser Pseudozufallszahlen verwendete Startparameter in regelmäßigen oder unregelmäßige Intervallen geändert wird.

[0113] **Fig.** 15 zeigt ein Beispiel der Konfiguration eines verschlüsselten Kommunikationssystems mit der Fähigkeit, den zum Generieren von Pseudozufallszahlen verwendeten Startparameter zu ändern. In **Fig.** 15 entsprechen die Arbeitsstationen **50** und **54** dem Server **10** bzw. dem Client **15** in **Fig.** 1. Die Anwendungsprogramme **51** und **55** entsprechen den Anwendungsprogrammen **13** bzw. **16** in **Fig.** 1. Die vertrauenswürdigen Agenten **52** und **56** entsprechen den vertrauenswürdigen Agenten **14** bzw. **17** in **Fig.** 1.

[0114] Der vertrauenswürdige Agent **52** hat eine Anwendungsschnittstelleneinheit **53**, und, wie mit Bezugnahme auf **Fig.** 7B erläutert wurde, einen Exklusiv-ODER-Generator **40**, einen Pseudozufallszahlengenerator **41**, eine Startparametersektion **42** und eine Startparametergeneriersektion (Startparameteränderungssektion) **43**. Der vertrauenswürdige Agent **56** hat eine Anwendungsschnittstelleneinheit **57**, und einen Exklusiv-ODER-Generator **40'**, einen Pseudozufallszahlengenerator **41'** und eine Startparametersektion **42'**. Der Exklusiv-ODER-Generator **40**, der Pseudozufallszahlengenerator **41** und die Startparametersektion **42**, und der Exklusiv-ODER-Generator **40'** und die Startparametersektion **42'** sind jeweils derselbe Typ von Einheiten.

[0115] Die Aktion des in **Fig.** 15 gezeigten verschlüsselten Kommunikationssystems ist wie folgt. Zuerst wird der vertrauenswürdige Agent **56**, der in mobilem Code beschrieben ist, von der Arbeitsstation **50** zur Arbeitsstation **54** transferiert. Als Nächstes schafft die Startparametergeneriersektion **43** 1 Startparameter und transferiert diesen Startparameter zu den Startparametersektionen **42** bzw. **42'**. Zu dieser Zeit wird ein verschlüsselter Kommunikationsweg zwischen den vertrauenswürdigen Agenten **52** und **56** hergestellt.

[0116] Die Startparametersektionen **42** und **42'** geben die Startparameter, die sie empfangen haben, in die Pseudozufallszahlengeneratoren **41** bzw. **41'** ein. Das heißt, die in die Pseudozufallszahlengeneratoren **41** und **41'** eingegebenen Startparameter gleichen einander. Die Pseudozufallszahlengeneratoren **41** und **41'** generieren Pseudozufallszahlen in Übereinstimmung mit den jeweiligen empfangenen Startparametern und geben sie in die Exklusiv-ODER-Generatoren **40** und **40'** ein. Die in den beiden Einheiten zu dieser Zeit generierten Pseudozufallszahlen gleichen einander. Danach generiert die Startparametergeneriersektion **43** neue Startparameter in regelmäßigen oder unregelmäßigen Intervallen in Übereinstimmung mit dem spezifizierten Algorithmus und transferiert sie zu den Startparametersektionen **42** und **42'**. Demgemäß werden dieselben Pseudozufallszahlen in den vertrauenswürdigen Agenten **52** und **56** generiert.

[0117] Daten von dem Anwendungsprogramm **51** werden zum Exklusiv-ODER-Generator **40** über die Anwendungsschnittstelleneinheit **53** gesendet. Dort werden die Daten unter Verwendung der vom Pseudozufallszahlengenerator **41** generierten Pseudozufallszahlen verschlüsselt. Die verschlüsselten Daten werden über das Internet **58** zur Arbeitsstation **54** gesendet. Die von der Arbeitsstation **54** empfangenen Daten werden in Übereinstimmung mit den vom Pseudozufallszahlengenerator **41'** generierten Pseudozufallszahlen im Exklusiv-ODER-Generator **40'** entschlüsselt. Dann werden diese entschlüsselten Daten über die Anwendungsschnittstelleneinheit **57** zum Anwendungsprogramm **55** transferiert.

[0118] In der oben beschriebenen Verschlüsselungs/Entschlüsselungsverarbeitung ist die Aktion des Gene-

rierens der Pseudozufallszahlen gegenseitig in den vertrauenswürdigen Agenten **52** und **56** synchronisiert, so dass in dem vertrauenswürdigen Agenten **52** verschlüsselte Daten in dem vertrauenswürdigen Agenten **56** entschlüsselt werden.

[0119] **Fig.** 16 zeigt ein Konfigurationsbeispiel einer weiteren Ausführungsform dieser Erfindung. In dem in **Fig.** 16 gezeigten System hat der vertrauenswürdige Agent **52** keine Startparametergeneriersektion **43**; Startparameter werden in einem Startparameter-Server **59** generiert, der mit dem Internet **58** verbunden ist. Der Startparameter-Server **59** hat Funktionen, die zu jenen der Startparametergeneriersektion **43** in **Fig.** 15 äquivalent sind; neue Startparameter werden in regelmäßigen oder unregelmäßigen Intervallen generiert und zu den Startparametersektionen **42** und **42'** transferiert.

[0120] Die Aktion dieses Systems ist wie folgt. Zuerst wird der vertrauenswürdige Agent **56**, der in mobilem Code beschrieben ist, von der Arbeitsstation **50** zur Arbeitsstation **54** transferiert. Als Nächstes fordert die Startparametersektion **42** der Arbeitsstation **50** vom Startparameter-Server **59** einen Startparameter an, der für eine verschlüsselte Kommunikation notwendig ist. Zu dieser Zeit benachrichtigt der vertrauenswürdige Agent **52** den Startparameter-Server **59** über das entsprechende Terminal der verschlüsselten Kommunikation (hier die Arbeitsstation **54**). Der Startparameter-Server **59** generiert einen Startparameter in Übereinstimmung mit dieser Anforderung, und transferiert den generierten Startparameter zu den Startparametersektionen **42** und **42'**. Dies sichert einen verschlüsselten Kommunikationsweg. Der Startparameter-Server **59** generiert anschließend neue Startparameter in regelmäßigen oder unregelmäßigen Intervallen und transferiert sie zu den Startparametersektionen **42** und **42'**. Die übrige Aktion ist gleich wie die in **Fig.** 15 beschriebene.

[0121] **Fig.** 17 zeigt ein Beispiel der Konfiguration einer weiteren Ausführungsform dieser Erfindung. In dem in **Fig.** 17 gezeigten System haben die vertrauenswürdigen Agenten **52** und **56** die Startparametergeneriersektionen **43** bzw. **43'**.

[0122] Die Startparametergeneriersektionen **43** und **43'** haben einander gleiche Funktionen. Zusätzlich sind die Aktionen der Startparametergeneriersektionen **43** und **43'** miteinander synchronisiert. Das heißt, dieselben Initialwerte werden in den Startparametergeneriersektionen **43** und **43'** eingestellt, und anschließend geben sie dieselben Startparameter sequentiell aus. Die Synchronisation zwischen den Startparametergeneriersektionen **43** und **43'** ist grundsätzlich gleich wie die Synchronisation zwischen den in **Fig.** 6 gezeigten Verschlüsselungsverfahrensauswahl-Steuereinheiten **25** und **39**.

[0123] Die Aktion des in **Fig.** 17 gezeigten Systems ist wie folgt. Zuerst wird der vertrauenswürdige Agent **56**, der in mobilem Code beschrieben ist, von der Arbeitsstation **50** zur Arbeitsstation **54** transferiert. Als Nächstes werden die von den Startparametergeneriersektionen **43** und **43'** generierten Startparameter zu den Startparametersektionen **42** bzw. **42'** transferiert, wodurch ein verschlüsselter Kommunikationsweg gesichert wird. Zu dieser Zeit geben die Startparametergeneriersektionen **43** und **43'** dieselben Startparameter in derselben Reihenfolge aus. Andere Aktionen sind gleich wie mit Bezugnahme auf **Fig.** 15 erläutert.

[0124] **Fig.** 18 ist ein Flussdiagramm, das die Aktion einer Startparametergeneriersektion oder eines Startparameter-Servers erläutert. Diese Verarbeitung ist grundsätzlich gleich wie jene in dem Flussdiagramm in **Fig.** 11, wo das Verschlüsselungsverfahren ausgewählt wird.

[0125] Der Initialstartparameter wird in Schritt S11 eingestellt. Das Verfahren zur Einstellung des Initialstartparameters ist wie mit Bezugnahme auf **Fig.** 11 erläutert wurde. In Schritt S12 werden Pseudozufallszahlen unter Verwendung des Pseudozufallszahlengenerators auf der Basis dieses Initialstartparameters generiert. In Schritt S13 und S14 werden die generierten Zufallszahlen zur Startparametersektion als geeignete Startparameter gesendet. In Schritt S15 wird die Zeiteinstellung bestimmt, zu welcher der Startparameter geändert wird. Diese Änderungszeiteinstellung wird beispielsweise durch einen Parameter wie die Anzahl von Paketen oder die Zeit angezeigt. In Schritt S16 wird überwacht, ob die Zeit die Startparameteränderungszeiteinstellung erreicht hat oder nicht. Wenn die Zeiteinstellung zur Änderung des Startparameters erreicht ist, wird in Schritt S17 die unmittelbar vorhergehende generierte Pseudozufallszahl als neuer Startparameter eingestellt, und die Prozedur kehrt zu Schritt S12 zurück. Danach werden die Schritte S12 bis S17 wiederholt. Der Startparameter wird gemäß der oben beschriebenen Verarbeitung in unregelmäßigen Intervallen geändert.

[0126] In dem in **Fig.** 15 gezeigten System führt die Startparametergeneriersektion **43** die oben beschriebene Verarbeitung durch. In dem in **Fig.** 16 gezeigten System führt der Startparameter-Server **59** die oben beschriebene Verarbeitung durch. In dem in **Fig.** 17 gezeigten System führen die Startparametergeneriersektionen **43** bzw. **43'** die oben beschriebene Verarbeitung durch. In dem in **Fig.** 17 gezeigten System haben der Exklu-

siv-ODER-Generator **40**, der Pseudozufallszahlengenerator **41**, die Startparametersektion **42** und die Startparametergeneriersektion **43**; und der Exklusiv-ODER-Generator **40'**, der Pseudozufallszahlengenerator **41'**, die Startparametersektion **42'** und die Startparametergeneriersektion **43'** jeweils einander gleiche Funktionen, so dass durch das Einstellen desselben Initialstartparameters in den Startparametergeneriersektionen **43** und **43'** dieselben Pseudozufallszahlen anschließend in derselben Reihenfolge generiert werden.

[0127] **Fig.** 19 ist eine Konfigurationsdarstellung für den Fall, in dem das verschlüsselte Kommunikationssystem dieser Ausführungsform an das WWW (World Wide Web) angepasst wird.

[0128] Die Software auf der Serverseite besteht aus dem WWW-Server **60**, dem permanent residenten vertrauenswürdigen Agenten **61** und dem Applet **62**, in das kryptographische Verarbeitungseinheiten eingebaut wurden. Das Applet **62** ist ein Agent mit kryptographischer Verarbeitung. Demgegenüber ist die Software auf der Clientseite der WWW-Browser **63**. Der vertrauenswürdige Agent **61** entspricht dem vertrauenswürdigen Agenten **14** in **Fig.** 1. Das Applet **62** ist in mobilem Code beschrieben und entspricht dem vertrauenswürdigen Agenten **11** in **Fig.** 1.

[0129] Die Aktion dieses Systems ist wie folgt. Zuerst wird, in dem WWW-Browser **63** auf der Clientseite, ein Zugriff auf den WWW-Server **60** vorgenommen; dann wird das Applet **62**, in dem die kryptographischen Verarbeitungseinheiten eingebaut sind, von der Serverseite zur Clientseite transferiert, und dieses Applet **62** wird in den WWW-Browser **63** eingebaut. Der Vorgang, durch den ein verschlüsselter Kommunikationsweg zwischen dem vertrauenswürdigen Agenten **61** und dem Applet **62** hergestellt wird, ist beispielsweise wie in **Fig.** 4 gezeigt.

[0130] Wenn der WWW-Browser **63** die gewünschten Daten vom WWW-Server **60** anfordert, werden die von dem WWW-Server **60** ansprechend auf diese Anforderung gesendeten Daten vom vertrauenswürdigen Agenten **61** verschlüsselt, und zur Clientseite gesendet. Auf der Clientseite wird der chiffrierte Text, der über den verschlüsselten Kommunikationsweg **64** transferiert wurde, von dem Applet **62** empfangen. Das Applet **62** kennt ein Verfahren zur Entschlüsselung der Daten, die von dem vertrauenswürdigen Agenten **61** verschlüsselt wurden. Das Applet **62** entschlüsselt den vom WWW-Server **60** empfangenen chiffrierten Text, und transferiert diese entschlüsselten Daten zur Browser-Software des WWW-Browsers **63**.

[0131] Somit werden vom WWW-Server **60** zum WWW-Browser **63** gesendete Daten von dem vertrauenswürdigen Agenten **61** verschlüsselt, bevor sie gesendet werden, und werden dann von dem Applet **62** entschlüsselt und reproduziert.

[0132] **Fig.** 20 zeigt ein Beispiel der Konfiguration in dem Fall, in dem die verschlüsselte Kommunikation dieser Erfindung einem Videotransmissionssystem oder einem Audiotransmissionssystem entspricht. In diesem Beispiel wird ein vertrauenswürdiger Agent, in den die kryptographischen Verarbeitungseinheiten eingebaut sind (ein vertrauenswürdiger Agent mit kryptographischer Verarbeitung), in Kombination mit Anwendungen für eine Videotransmission und Audiotransmission verwendet.

[0133] Die Arbeitsstation **70**, von der Audiodaten und Videodaten gesendet werden, besteht aus der Kamera **71**, dem Analog/Digital(A/D)-Wandler **72**, dem Rahmenpuffer **73**, dem Mikrophon **74**, dem Analog/Digital(A/D)-Wandler **75**, dem Puffer **76** und dem vertrauenswürdigen Agenten **77** des permanent residenten Typs, der eine Videodaten/Audiodaten-Verschlüsselungsfunktion aufweist. Zusätzlich hat sie den vertrauenswürdigen Agenten **78**, in dem die Entschlüsselungsfunktion, die der Verschlüsselungsverarbeitung in dem vertrauenswürdigen Agenten **77** entspricht, in mobilem Code beschrieben ist.

[0134] Die Arbeitsstation **80**, welche die Videodaten/Audiodaten empfängt, besteht aus dem vertrauenswürdigen Agenten **78**, in den kryptographische Verarbeitungseinheiten eingebaut sind, und der von der sendeseitigen Arbeitsstation **70** gesendet wird; dem Rahmenpuffer **82**; dem Digital/Analog(D/A)-Wandler **83**; der Anzeige **84**; dem Audiodaten-Empfangspuffer **85**; dem Digital/Analog(D/A)-Wandler **86** und dem Lautsprecher **87**.

[0135] Die Aktion in diesem System, wenn Videodaten gesendet und empfangen werden, ist wie folgt. Zuerst wird eine Anforderung, den vertrauenswürdigen Agenten **78** zu senden, von der Arbeitsstation **80**, welche die Videodaten empfangen will, an die sendeseitige Arbeitsstation **70** gesendet. Wenn die sendeseitige Arbeitsstation **70** diese Sendeanforderung empfängt, sendet sie den vertrauenswürdigen Agenten **78**, der benötigt wird, wenn Bilddaten entschlüsselt werden, an die empfangsseitige Arbeitsstation **80**. Dies vollendet die Vorbereitung für einen Datentransfer.

[0136] Die sendeseitige Arbeitsstation **70** konvertiert die von der Kamera **71** aufgenommenen Bilddaten mittels des Analog/Digital(A/D)-Wandlers **72** in ein Digitalsignal und sendet es zu dem Rahmenpuffer **73**. Der Rahmenpuffer **73** speichert die Daten aus dem Analog/Digital(A/D)-Wandler **72**, um die Differenzen zwischen der Rate, bei der Daten von der Kamera **71** eingegeben werden, und der Verschlüsselungsverarbeitungsrate in dem vertrauenswürdigen Agenten **77** zu absorbieren.

[0137] Als Nächstes werden die Ausgangsdaten vom Rahmenpuffer **73** durch den vertrauenswürdigen Agenten **77** verschlüsselt und in das Netz ausgesendet. In der empfangsseitigen Arbeitsstation **80** werden die verschlüsselten Bilddaten, die gesendet wurden, empfangen und durch den vertrauenswürdigen Agenten **78** entschlüsselt. Die entschlüsselten Bilddaten werden durch den Digital/Analog(D/A)-Wandler **83** über den empfangsseitigen Rahmenpuffer **82** in ein Analogsignal rückkonvertiert und auf der Anzeige **84** angezeigt.

[0138] In dem in **Fig.** 20 gezeigten System ist die Aktion in dem Fall der Transmission von Audiodaten nahezu gleich wie die oben beschriebene. Das heißt, die Echtzeitdaten, die transferiert werden, sind Audiodaten anstelle von Videodaten, das Mikrophon **74** ersetzt die Kamera **71** als Eingabesektion für die zu transferierenden Daten, und der Lautsprecher **87** ersetzt die Anzeige **84** als Ausgabesektion. Ansonsten ist die Aktion grundsätzlich gleich.

[0139] **Fig.** 21 ist eine Konfigurationsdarstellung für den Fall, in dem die verschlüsselte Kommunikation dieser Erfindung in einem elektronischen Konferenzsystem verwendet wird. In dem elektronischen Konferenzsystem dieser Ausführungsform sind eine Agentenverteilungsstation **90** und eine Vielzahl von Hosts **91** bis **94** über ein Netz **95** miteinander verbunden. Die Agentenverteilungsstation **90** hat eine Benutzererkennungsfunktion und verteilt Agenten ansprechend auf Anforderungen offizieller Benutzer. Das Netz **95** ist beispielsweise ein LAN. Der Multicast-Kommunikationsweg **96** ist ein Transmissionsweg zum Senden und Empfangen von Daten unter den Hosts **91** bis **94** während einer elektronischen Konferenz. Der Multicast-Kommunikationsweg **96** kann innerhalb des Netzes **95** hergestellt werden, oder er kann auf anderen physischen Leitungen getrennt vom Netz **95** hergestellt werden.

[0140] Ein Host, der an einer elektronischen Konferenz teilnimmt, fordert von der Agentenverteilungsstation **90** an, einen Agenten zu senden, der für eine verschlüsselte Kommunikation benötigt wird, um einen verschlüsselten Kommunikationsweg herzustellen. Das heißt, wenn beispielsweise die Hosts **91** bis **94** an einer elektronischen Konferenz teilnehmen, übermittelt einer unter diesen Hosts ein Mitglied, das an der elektronischen Konferenz teilnimmt, an die Agentenverteilungsstation **90**. Die Agentenverteilungsstation **90** sendet dann einen vertrauenswürdigen Agenten, in dem kryptographische Verarbeitungseinheiten eingebaut sind, an den Host, der die Anforderung gestellt hat. Die vertrauenswürdigen Agenten **97**, die an die jeweiligen Hosts **91** bis **94** verteilt werden, sichern verschlüsselte Kommunikationswege unter diesen Hosts unter Verwendung des Multicast-Kommunikationswegs **96**. Anschließend werden Daten, die sich auf die elektronische Konferenz beziehen, in verschlüsselter Form gesendet und empfangen.

[0141] Die Agentenverteilungsstation **90** kann so konfiguriert sein, dass sie auch die Funktion(en) des Verschlüsselungsverfahrensinstruktions-Servers und/oder des Verschlüsselungsverarbeitungseinheiten-Verteilungsservers erfüllt, die mit Bezugnahme auf **Fig.** 9 beschrieben sind, oder beispielsweise des Startparameter-Servers **59**, der in **Fig.** 16 gezeigt ist.

[0142] Als Nächstes wird die Schnittstelle zwischen dem vertrauenswürdigen Agenten und der Anwendung erläutert. Hier wird, wie in **Fig.** 22A gezeigt, ein Fall behandelt, in dem Daten zwischen der Informationsverarbeitungseinheit **10** (dem Server in **Fig.** 1) und der Informationsverarbeitungseinheit **15** (dem Client in **Fig.** 1) gesendet und empfangen werden. In diesem Fall wird, unter den Einstellungen des Anwendungsprogramms **13** in der Informationsverarbeitungseinheit **10**, die Informationsverarbeitungseinheit **15** als Kommunikationspartner spezifiziert, und ein Port, durch den das Anwendungsprogramm **16** Daten empfängt, wird als Kommunikationsport spezifiziert. Ähnlich wird, in den Einstellungen des Anwendungsprogramms **16** der Informationsverarbeitungseinheit **15**, die Informationsverarbeitungseinheit **10** als Kommunikationspartner spezifiziert, und ein Port, durch den das Anwendungsprogramm **13** Daten empfängt, wird als Kommunikationsport spezifiziert.

[0143] Um eine verschlüsselte Kommunikation vorzunehmen, wie in **Fig.** 22B gezeigt, wenn die vertrauenswürdigen Agenten **14** und **17** verwendet werden, wird in den Einstellungen des Anwendungsprogramms **13** dieselbe Einheit (die Informationsverarbeitungseinheit **10**) als Kommunikationspartner spezifiziert, und ein Port, durch den der vertrauenswürdige Agent **14** Daten empfängt, wird als Kommunikationsport spezifiziert. Ähnlich wird, in den Einstellungen des Anwendungsprogramms **17**, dieselbe Einheit (die Informationsverarbeitungseinheit **15**) als Kommunikationspartner spezifiziert, und ein Port, durch den der vertrauenswürdige Agent

17 Daten empfängt, wird als Kommunikationsport spezifiziert.

[0144] Somit werden durch das Ändern der Einstellungen beispielsweise der Kommunikationsports Daten, die zwischen den Anwendungsprogrammen **13** und **16** gesendet und empfangen werden, über die vertrauenswürdigen Agenten **14** und **17** gesendet. Das heißt, die von den Anwendungsprogrammen **13** und **14** gesendeten und empfangenen Daten können verschlüsselt werden, indem nur Einstellungen wie die Kommunikationsports geändert werden, ohne die Anwendungsprogramme **13** und **16** selbst zu ändern.

[0145] Die Proxy (Kommunikationsleitwegport)-Einstellung wird grundsätzlich auf dieselbe Weise geändert wie der oben beschriebene Kommunikationspartner und Kommunikationsport. Das heißt, wenn es eine Funktion zum Einstellen des Proxy in einem Anwendungsprogramm gibt, werden die Informationsverarbeitungseinheit, in der das Anwendungsprogramm installiert ist, und ein Port, durch den der in dieser Informationsverarbeitungseinheit installierte vertrauenswürdige Agent Daten empfängt, als Proxy eingestellt.

[0146] Die vom vertrauenswürdigen Agenten vorgesehene API (Application Interface – Anwendungsschnittstelle) wird anstelle der vom System vorgesehenen API verwendet. In diesem Fall ist es normalerweise notwendig zu rekompilieren, nachdem das Quellprogramm geändert wird. Wenn es beispielsweise keinen vertrauenswürdigen Agenten gibt, wird die Sektion geändert, in der "open( );" auftritt, wohingegen, wenn es einen vertrauenswürdigen Agenten gibt, die Sektion "openTrusted( );" geändert werden sollte, und dann wird das Quellprogramm rekompiliert.

[0147] Es ist auch möglich, einen vertrauenswürdigen Agenten dieser Ausführungsform als Kernmodul des Betriebssystems (OS) zu realisieren, und in das OS, wie notwendig, einzubauen. In **Fig.** 22B ist es beispielsweise auch möglich, dass der vertrauenswürdige Agent **14** auf der Kernebene des OS eingebaut wird, das in der Informationsverarbeitungseinheit **10** installiert ist.

[0148] Ein vertrauenswürdiger Agent, wie in **Fig.** 3 gezeigt, hat eine Anwendungsschnittstellensektion und eine kryptographische Verarbeitungssektion. Die kryptographische Verarbeitungssektion besteht, wie in **Fig.** 4 gezeigt, aus einer Sendesektion und einer Empfangssektion. Die Sendesektion hat eine Datenverschlüsselungsfunktion, wohingegen die Empfangssektion eine Funktion zum Entschlüsseln eines chiffrierten Texts aufweist. Wenn in dieser Ausführungsform ein vertrauenswürdiger Agent, der in mobilem Code beschrieben ist, von dem Server an den Client gesendet wird, ist es möglich, nur die Anwendungsschnittstellensektion und die Sendesektion zu senden, oder nur die Anwendungsschnittstellensektion und die Empfangssektion.

[0149] Eine Anwendung zum Verschlüsseln und Senden von Daten ist die Fernsehkommunikation, wie VOD (Video on Demand – Videoabruf). Bei der Fernsehkommunikation muss die empfangsseitige Einheit keine Funktion zur Verschlüsselung von Daten als Codeverarbeitungsfunktion haben; sie muss nur eine Funktion zur Entschlüsselung des chiffrierten Texts haben, der gesendet wird. Wenn in diesem Fall ein vertrauenswürdiger Agent zur Fernsehkommunikations-Empfangseinheit gesendet wird, werden demgemäß nur die Anwendungsschnittstellensektion und die Empfangssektion gesendet.

[0150] Ein Beispiel eines Programms eines vertrauenswürdigen Agenten, der in mobilem Code beschrieben ist, ist in **Fig.** 23 bis **Fig.** 29 gezeigt. Dieses Programm entspricht dem vertrauenswürdigen Agenten **11** in **Fig.** 1, und wird zum Client transferiert. Dieses Programm enthält eine Funktion zur Ausführung der kryptographischen Verarbeitung von Bilddaten ein.

[0151] Dieses Programm dient dem Zweck des Einlesens von Dateien von einem WWW (World Wide Web)-Server und Anzeigens animierter Bilder. Es ist in Java beschrieben (einer objektorientierten Sprache für die Verwendung im Internet, entwickelt von Sun Microsystems). Dieses Programm hat auch eine Funktion zum Einlesen von Bildinformationen im Bitmap-Format (T0 bis T9.ppm) von einem Server, 2.048 Bytes (b[ ]) zu einem Zeitpunkt, und Vornehmen von Applet-Anzeigen. Dieses Programm zeigt 10 Bilddateien eine nach der anderen an und wiederholt dann die Aktion. Ein Überblick über das Anzeigeverfahren ist wie folgt. Zuerst wird ein Kommunikationsweg zu und von dem Server hergestellt, und die notwendigen Bilddateien werden angefordert. Als Nächstes wird dieser Kommunikationsweg verwendet, um Bilddateien zu empfangen, und Bilder werden angezeigt.

[0152] Nun erfolgt eine detailliertere Erläuterung des Programms mit Bezugnahme auf die Zeichnungen.

(A) Zuerst wird der Klassenweg definiert.

(B) Die zu verwendenden Variablen werden bestimmt.

(C) Die init-Funktion ist eine Funktion, die Initialeinstellungen für den Zweck der Initiation der Kommunika-

tion mit dem Server vornimmt. Der Name des Servers, mit dem eine Verbindung erfolgt, die Serverportnummer und notwendige Dateinamen werden spezifiziert. Zusätzlich werden die Pseudozufallszahlen-Generierspezifikationen bestimmt.
(D) Die makesocket-Funktion wird verwendet, um eine Verbindungsanforderung zum Server zu senden und einen Kommunikationsweg zu schaffen.
(E) Der Kommunikationsweg, der geschaffen wurde, wird verwendet, um Informationen bezüglich der Bildbreite und -höhe von dem Server zu erhalten. Diese Informationen sind notwendig, wenn Bilder auf der Clientseite reproduziert werden.
(F) Als makesocket-Funktionsverarbeitung wird eine Verbindungsanforderung an den Server gesendet, und die Verarbeitung zur Herstellung eines Kommunikationswegs wird beschrieben.
(G) Die sendimagefile-Funktion ist eine Funktion, die den Kommunikationsweg verwendet, um die notwendigen Bilddateinamen an den Server zu senden.
(H) Die getimage-Funktion ist eine Funktion, die den Kommunikationsweg verwendet, um Bilddateien vom Server zu empfangen und Bilder zu schaffen.
(I) Wenn es in den Bilddateien keine Daten mehr gibt, endet die Verarbeitung.
(J) Bilddaten werden entschlüsselt, 1 Byte zu einem Zeitpunkt.
(K) Ein Pixel wird jedes 4. Byte geschaffen. Ein Pixel hat 4 Komponenten: Helligkeit, rot, grün und blau.
(L) Bilder werden aus Pixeln aufgebaut.
(M) Die Verschlüsselung wird vorgeschrieben.
(N) Wenn das Programm hochfährt, wird dieses Programm als Thread ausgeführt. Durch das Ausführen als Thread wird es möglich, eine Vielzahl von Verarbeitungen parallel innerhalb eines Programms auszuführen.
(O) Die Thread-Aktion wird vorgeschrieben. Nachdem ein Kommunikationsweg durch die init-Funktion geschaffen wird, wird die folgende substantive Verarbeitung tatsächlich ausgeführt.
(P) Die Verarbeitung zur Anzeige von 10 Bilddateien wird vorgenommen; dann wird dies wiederholt.
(Q) Der Kommunikationsweg wird verwendet, um die notwendigen Bilddateinamen an den Server zu senden. Die sendimagefile-Funktion wird für diese Verarbeitung verwendet.
(R) Der Kommunikationsweg wird verwendet, um Bilddateien vom Server zu empfangen und Bilder zu schaffen. Die getimage-Funktion wird verwendet.
(S) Die geschaffenen Bilder werden angezeigt.

[0153] Diese Erfindung macht eine verschlüsselte Kommunikation möglich, wie oben erläutert, indem ein Agent, der kryptographische Verarbeitungseinheiten enthält, an den Kommunikationspartner gesendet wird, mit dem eine verschlüsselte Kommunikation durchzuführen ist, oder indem ein Agent, der kryptographische Verarbeitungseinheiten enthält, von diesem Partner empfangen wird. Aus diesem Grund ist es möglich, eine Veröffentlichung des Verschlüsselungsverfahrens zu vermeiden; und durch die Verwendung von Agenten kann das Verschlüsselungsverfahren in regelmäßigen oder unregelmäßigen Intervallen geändert werden, und die bei der Verschlüsselung notwendigen Parameter können geändert werden, um es schwierig zu machen, die verschlüsselten Daten zu dechiffrieren. Demgemäß wird ein starkes Verschlüsselungsverfahren erhalten mit einem geringen Gesamtaufwand, das für eine Echtzeitkommunikation geeignet ist.

[0154] Diese Erfindung ist nicht auf ein Kryptosystem beschränkt, sondern kann verbreitet angewendet werden, um Systeme zu codieren/decodieren (Modulation/Demodulation). In diesem Fall wird ein Agent, der ein in mobilem Code beschriebenes Programm für eine Codier/Decodier(Modulations/ Demodulations)-Verarbeitung enthält, vor der Datentransmission gesendet.

Patentansprüche

1. Verschlüsselungskommunikationsverfahren zum Senden verschlüsselter Daten zwischen einem ersten Terminal und einem zweiten Terminal, welches die Schritte umfasst:
Senden, in einer verschlüsselten Weise, von dem ersten Terminal, in dem ein erster Agent installiert ist, der ein Programm zur kryptographischen Verarbeitung enthält, eines zweiten Agenten mit derselben Funktion wie der erste Agent an das zweite Terminal; und
Vornehmen einer verschlüsselten Kommunikation zwischen dem ersten Agenten und dem zweiten Agenten;
wobei das Verfahren ferner den Schritt umfasst:
unabhängiges und synchrones Ändern eines Parameters,
der für die verschlüsselte Kommunikation notwendig ist, wenn die kryptographische Verarbeitungssequenz die Zeit zum Umschalten des Verschlüsselungsverfahrens erreicht.

2. Verschlüsselungskommunikationsverfahren zum Senden verschlüsselter Daten zwischen einem ersten Terminal und einem zweiten Terminal, welches die Schritte umfasst:

Senden, in einer verschlüsselten Weise, von dem ersten Terminal, in dem ein erster Agent installiert ist, der ein Programm zur kryptographischen Verarbeitung enthält, eines zweiten Agenten mit derselben Funktion wie der erste Agent an das zweite Terminal; und
Vornehmen einer verschlüsselten Kommunikation zwischen dem ersten Agenten und dem zweiten Agenten;
wobei der erste und der zweite Agent jeweils eine Vielzahl kryptographischer Verarbeitungseinheiten vorsehen, und den weiteren Schritt:
Ändern der kryptographischen Verarbeitungseinheiten, die zu verwenden sind, während eine Synchronisation zwischen dem ersten und dem zweiten Agenten aufrechterhalten wird.

3. Verschlüsselungskommunikationsverfahren nach Anspruch 2,
bei welchem ein Verschlüsselungsverfahrensauswahl-Server, der jeden von dem ersten und dem zweiten Agenten anweist, welche kryptographische Verarbeitungseinheit zu verwenden ist, vorgesehen wird, wobei das Verfahren ferner die Schritte umfasst:
Anweisen, welche kryptographische Verarbeitungseinheit zu verwenden ist, von dem Verschlüsselungsverfahrensauswahl-Server an jeden von dem ersten und dem zweiten Agenten; und
Ändern, an jedem von dem ersten und dem zweiten Agenten, der zu verwendenden kryptographischen Verarbeitungseinheiten in Übereinstimmung mit der von dem Verschlüsselungsverfahrensauswahl-Server empfangenen Instruktion.

4. Verschlüsselungskommunikationsverfahren nach Anspruch 2,
bei welchem jeder von dem ersten und dem zweiten Agenten eine Funktion zur Bestimmung der zu verwendenden kryptographischen Verarbeitungseinheit hat, und die Funktionen gleich sind, wobei das Verfahren ferner die Schritte umfasst:
Einstellen desselben Initialwerts in jedem von dem ersten und dem zweiten Agenten;
Bestimmen einer zu verwendenden kryptographischen Verarbeitungseinheit in Übereinstimmung mit dem Initialwert in jedem von dem ersten und dem zweiten Agenten; und
Ändern kryptographischer Verarbeitungseinheiten gemäß der Bestimmung in jedem von dem ersten und dem zweiten Agenten.

5. Verschlüsselungskommunikationsverfahren nach einem der vorhergehenden Ansprüche, bei welchem durch den ersten und den zweiten Agenten vorgesehene Verschlüsselungsverfahren eine Pseudozufallszahl verwenden.

6. Verschlüsselungskommunikationsverfahren nach Anspruch 5, welches ferner die Schritte umfasst:
Schaffen eines Startparameters einer Pseudozufallszahl in dem ersten Terminal;
Einstellen des geschaffenen Startparameters in dem ersten und dem zweiten Agenten; und
Generieren von Pseudozufallszahlen in Übereinstimmung mit dem eingestellten Startparameter und Ausführen einer kryptographischen Verarbeitung unter Verwendung der Pseudozufallszahlen in dem ersten bzw. dem zweiten Agenten.

7. Verschlüsselungskommunikationsverfahren nach Anspruch 6, welches ferner den Schritt umfasst:
Ändern der kryptographischen Verarbeitungen in regelmäßigen oder unregelmäßigen Intervallen.

8. Verschlüsselungskommunikationsverfahren nach Anspruch 5,
bei welchem ein Startparameter-Server, der einen Startparameter für eine Pseudozufallszahl generiert, vorgesehen wird, wobei das Verfahren ferner die Schritte umfasst:
Generieren eines Startparameters für eine Pseudozufallszahl in dem Startparameter-Server;
Einstellen des generierten Startparameters in dem ersten und dem zweiten Agenten; und
Generieren einer Pseudozufallszahl in Übereinstimmung mit dem eingestellten Startparameter und Ausführen einer kryptographischen Verarbeitung unter Verwendung der Pseudozufallszahl in dem ersten bzw. dem zweiten Agenten.

9. Verschlüsselungskommunikationsverfahren nach Anspruch 2, 3 oder 4,
bei welchem jeder von dem ersten und dem zweiten Agenten eine Funktion zur Schaffung eines Startparameters für eine Pseudozufallszahl hat, wobei die Funktionen gleich sind,
wobei das Verfahren ferner die Schritte umfasst:
Einstellen desselben Initialwerts in jedem von dem ersten und dem zweiten Agenten;
Schaffen eines Startparameters für eine Pseudozufallszahl gemäß dem Initialwert in jedem von dem ersten und dem zweiten Agenten; und
Generieren einer Pseudozufallszahl in Übereinstimmung mit dem Startparameter und Ausführen einer krypto-

graphischen Verarbeitung unter Verwendung des Startparameters in dem ersten bzw. dem zweiten Agenten.

10. Verschlüsselungskommunikationsverfahren nach einem der vorhergehenden Ansprüche, bei welchem eine Vielzahl zweiter Terminals vorgesehen wird, und der Sendeschritt das Verteilen des zweiten Agenten an jedes der zweiten Terminals umfasst.

11. Verschlüsselungskommunikationsverfahren nach einem der vorhergehenden Ansprüche, bei welchem der zweite Agent in einem mobilen Code beschrieben wird.

12. Speichermedium, welches ein Programm speichert, das die folgenden Funktionen vorsieht, wenn es von einem Computer verwendet wird:
(a) Vornehmen einer kryptographischen Verarbeitung in einem Transferziel-Terminal,
(b) Senden, in einer verschlüsselten Weise, eines Agenten, der ein Programm enthält, das der Funktion (a) entspricht, an das Transferziel-Terminal, und
(c) Vornehmen einer verschlüsselten Kommunikation mit dem gesendeten Agenten;
mit der weiteren Funktion:
(d) Ändern eines Parameters, der für die verschlüsselte Kommunikation mit dem gesendeten Agenten notwendig ist, synchron mit und unabhängig von dem gesendeten Agenten, wenn die kryptographische Verarbeitungssequenz die Zeit zum Umschalten des Verschlüsselungsverfahrens erreicht.

13. Speichermedium, welches ein Programm speichert, das die folgenden Funktionen vorsieht, wenn es von einem Computer verwendet wird:
(a) Vornehmen einer kryptographischen Verarbeitung in einem Transferziel-Terminal,
(b) Senden, in einer verschlüsselten Weise, eines Agenten, der ein Programm enthält, das der Funktion (a) entspricht, an das Transferziel-Terminal, und
(c) Vornehmen einer verschlüsselten Kommunikation mit dem gesendeten Agenten;
mit den weiteren Funktionen:
(d) Versehen des Agenten mit einer Vielzahl kryptographischer Verarbeitungseinheiten, und
(e) Ändern der zu verwendenden kryptographischen Verarbeitungseinheiten, während eine Synchronisation zwischen dem Computer und dem Transferziel-Terminal aufrechterhalten wird.

14. Speichermedium, welches ein Programm speichert, nach Anspruch 12 oder 13, bei welchem die Funktion (b) angepasst ist, den Agenten an eine Vielzahl von Terminals zu senden, um so eine verschlüsselte Kommunikation zwischen der die Agenten verwendenden Vielzahl von Terminals vorzunehmen.

15. Verschlüsseltes Kommunikationssystem, welches zumindest einen Computer mit einem Speichermedium nach Anspruch 12, 13 oder 14 enthält.

16. Informationsverarbeitungseinheit (**10**) zur Verwendung bei einer verschlüsselten Kommunikation, welche enthält:
einen ersten Agenten (**14**), der in der Informationsverarbeitungseinheit (**10**) permanent resident ist, und der eine Einrichtung zur kryptographischen Verarbeitung von Informationen enthält;
einen zweiten Agenten (**11**), der dieselbe Funktion wie der erste Agent hat, zur Verwendung an einem Ziel der verschlüsselten Kommunikation; und
eine Sendeeinrichtung (**12**) zum Senden, in einer verschlüsselten Weise, des zweiten Agenten (**11**) an das Ziel;
wodurch, nach der Transmission des zweiten Agenten durch die Sendeeinrichtung, der erste Agent (**14**) in der Informationsverarbeitungseinheit (**10**) und der zweite Agent (**11**) am Ziel eine verschlüsselte Kommunikation vornehmen können;
wobei
der erste Agent und der zweite Agent unabhängig und synchron einen Parameter ändern, der für die verschlüsselte Kommunikation notwendig ist, wenn die kryptographische Verarbeitungssequenz die Zeit zum Umschalten des Verschlüsselungsverfahrens erreicht.

17. Informationsverarbeitungseinheit (**10**) zur Verwendung bei einer verschlüsselten Kommunikation, welche enthält:
einen ersten Agenten (**14**), der in der Informationsverarbeitungseinheit (**10**) permanent resident ist, und der eine Einrichtung zur kryptographischen Verarbeitung von Informationen enthält;
einen zweiten Agenten (**11**), der im Wesentlichen dieselbe Funktion wie der erste Agent hat, zur Verwendung an einem Ziel der verschlüsselten Kommunikation; und
eine Sendeeinrichtung (**12**) zum Senden, in einer verschlüsselten Weise, des zweiten Agenten (**11**) an das Ziel;

wodurch, nach der Transmission des zweiten Agenten durch die Sendeeinrichtung, der erste Agent (**14**) in der Informationsverarbeitungseinheit (**10**) und der zweite Agent (**11**) am Ziel eine verschlüsselte Kommunikation vornehmen können;
wobei der erste Agent (**14**) und der zweite Agent (**11**) jeweils eine Vielzahl kryptographischer Verarbeitungseinheiten vorsehen, und der erste und der zweite Agent eingerichtet sind, die zu verwendenden kryptographischen Verarbeitungseinheiten zu ändern, während eine Synchronisation zwischen dem ersten und dem zweiten Agenten aufrechterhalten wird.

Es folgen 29 Blatt Zeichnungen

Anhängende Zeichnungen

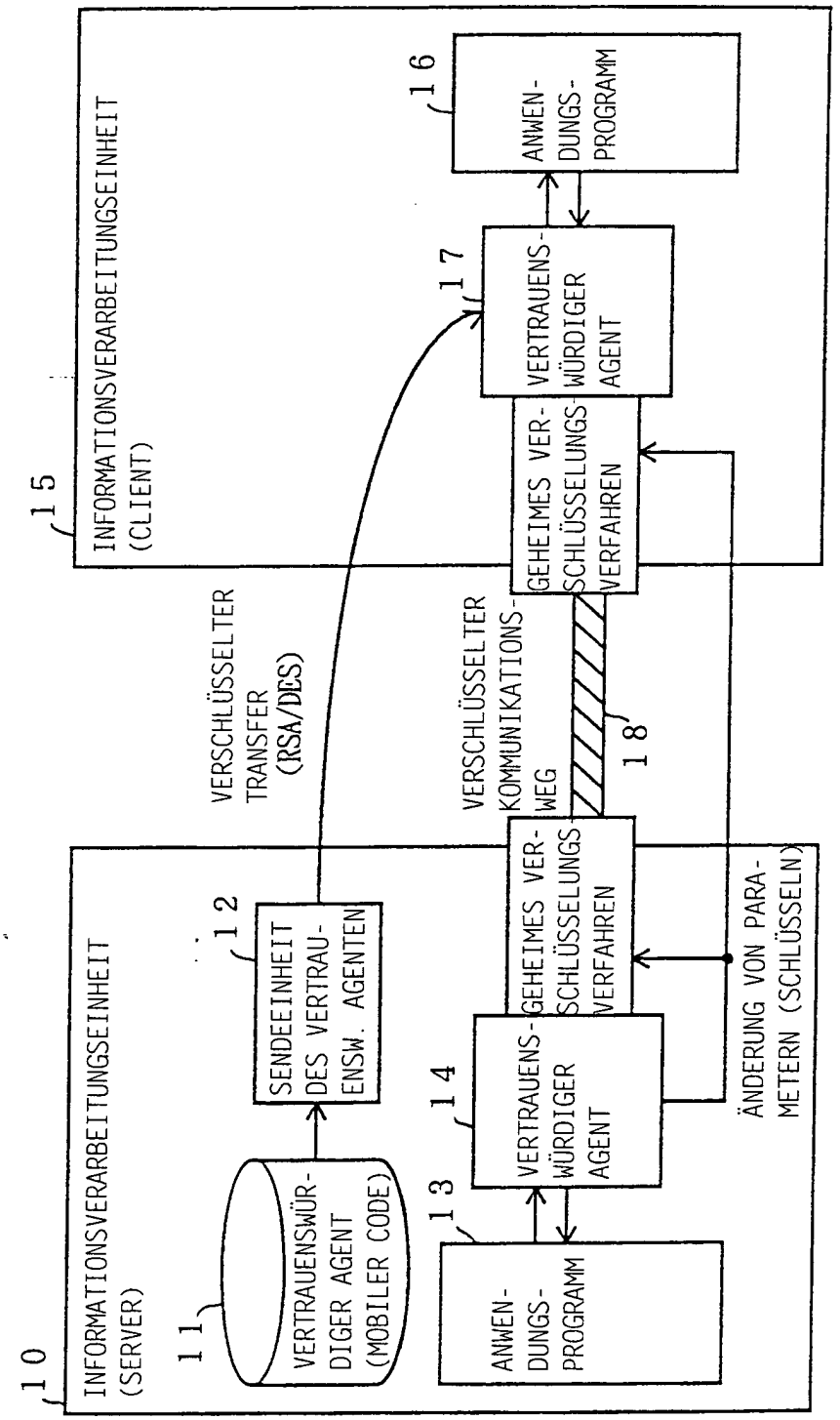


FIG. 1

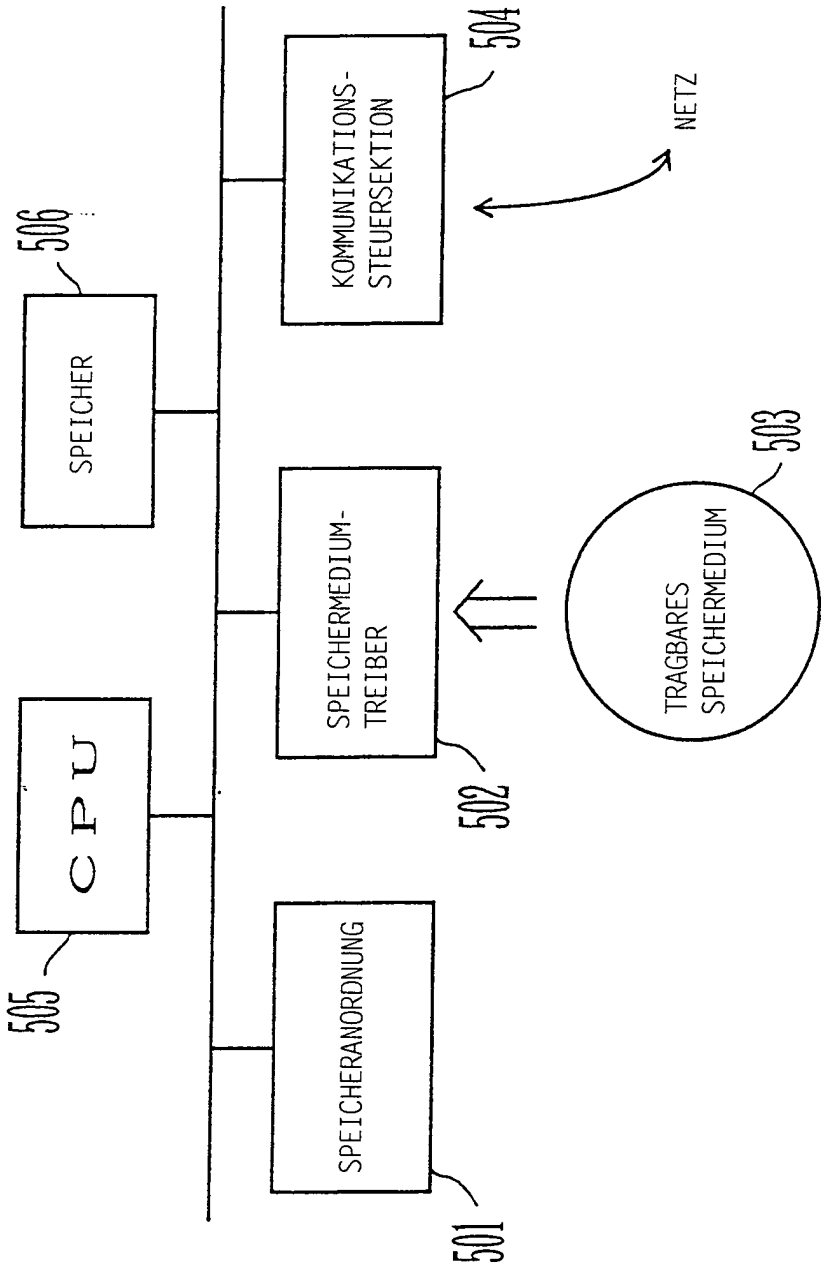

501
SPEICHERANORDNUNG
505
CPU
502
SPEICHERMEDIUM-
TREIBER
503
TRAGBARES
SPEICHERMEDIUM
504
KOMMUNIKATIONS-
STEUERSEKTION
NETZ
506
SPEICHER

FIG. 2

FIG. 3A

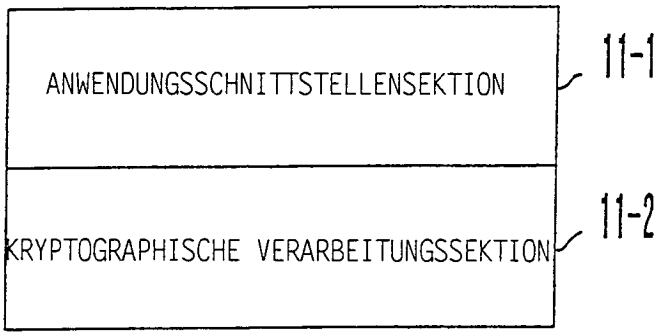
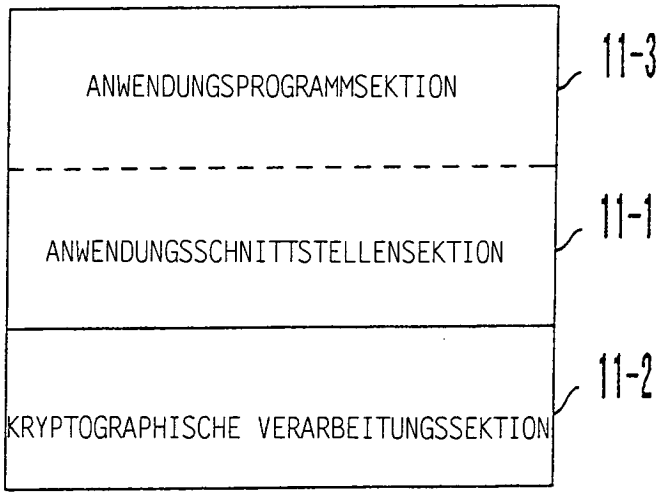

ANWENDUNGSSCHNITTSTELLENSEKTION
11-1
KRYPTOGRAPHISCHE VERARBEITUNGSSEKTION
11-2

FIG. 3B


ANWENDUNGSPROGRAMMSEKTION
11-3
ANWENDUNGSSCHNITTSTELLENSEKTION
11-1
KRYPTOGRAPHISCHE VERARBEITUNGSSEKTION
11-2

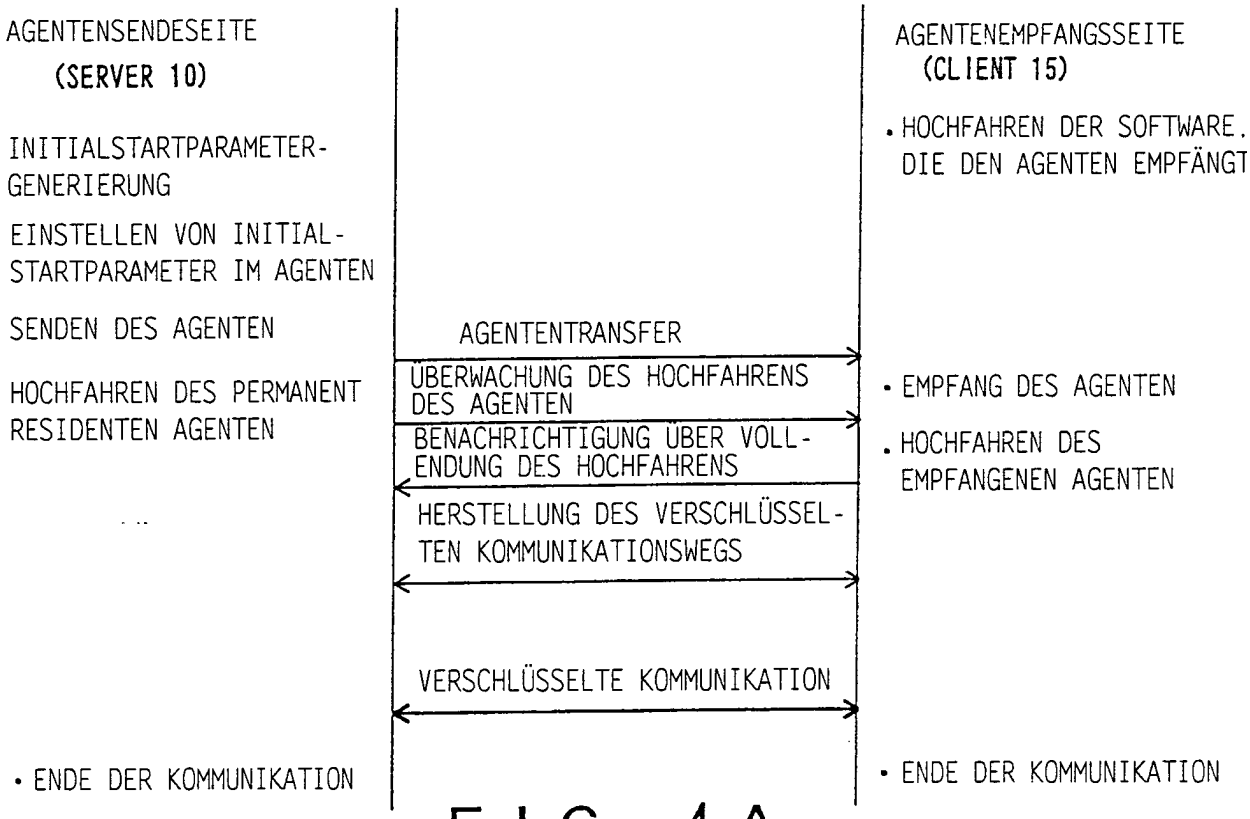

AGENTENSENDESEITE
(SERVER 10)
INITIALSTARTPARAMETER-GENERIERUNG
EINSTELLEN VON INITIAL-STARTPARAMETER IM AGENTEN
SENDEN DES AGENTEN
HOCHFAHREN DES PERMANENT RESIDENTEN AGENTEN
AGENTENEMPFANGSSEITE
(CLIENT 15)
HOCHFAHREN DER SOFTWARE, DIE DEN AGENTEN EMPFÄNGT
AGENTENTRANSFER
ÜBERWACHUNG DES HOCHFAHRENS DES AGENTEN
EMPFANG DES AGENTEN
BENACHRICHTIGUNG ÜBER VOLL-ENDUNG DES HOCHFAHRENS
HOCHFAHREN DES EMPFANGENEN AGENTEN
HERSTELLUNG DES VERSCHLÜSSEL-TEN KOMMUNIKATIONSWEGS
VERSCHLÜSSELTE KOMMUNIKATION
ENDE DER KOMMUNIKATION
ENDE DER KOMMUNIKATION

FIG. 4A

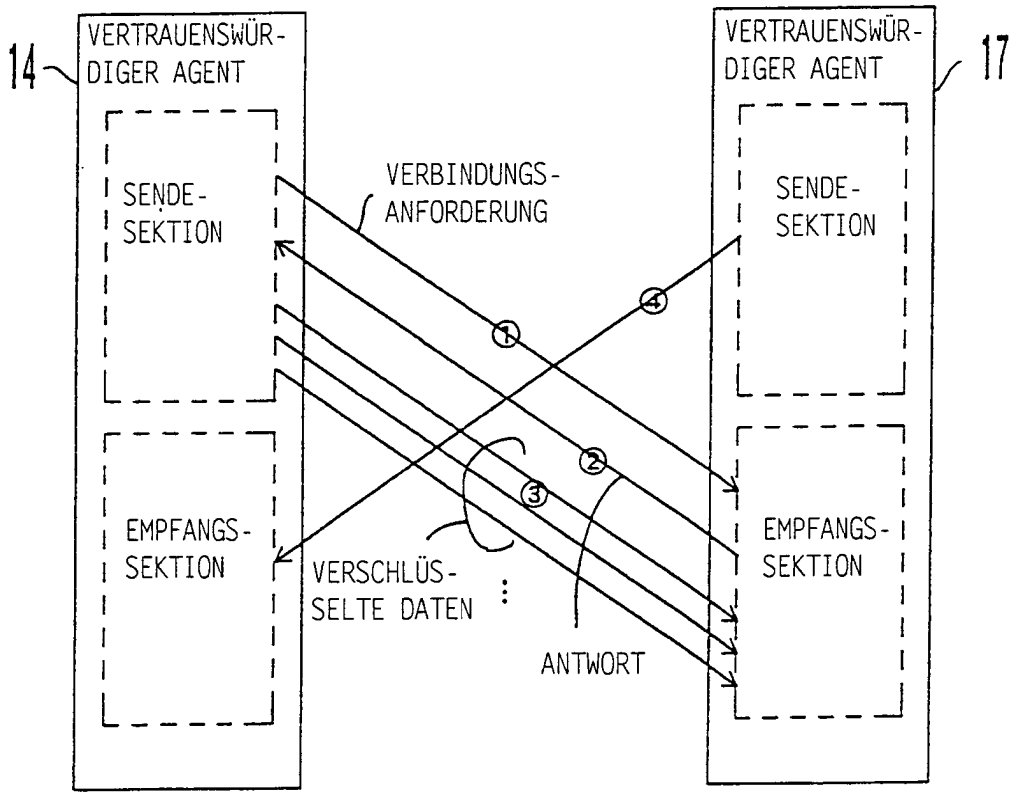

14
VERTRAUENSWÜR-DIGER AGENT
SENDE-SEKTION
EMPFANGS-SEKTION
VERBINDUNGS-ANFORDERUNG
VERSCHLÜS-SELTE DATEN
ANTWORT
VERTRAUENSWÜR-DIGER AGENT
17
SENDE-SEKTION
EMPFANGS-SEKTION
1
2
3
4

FIG. 4B

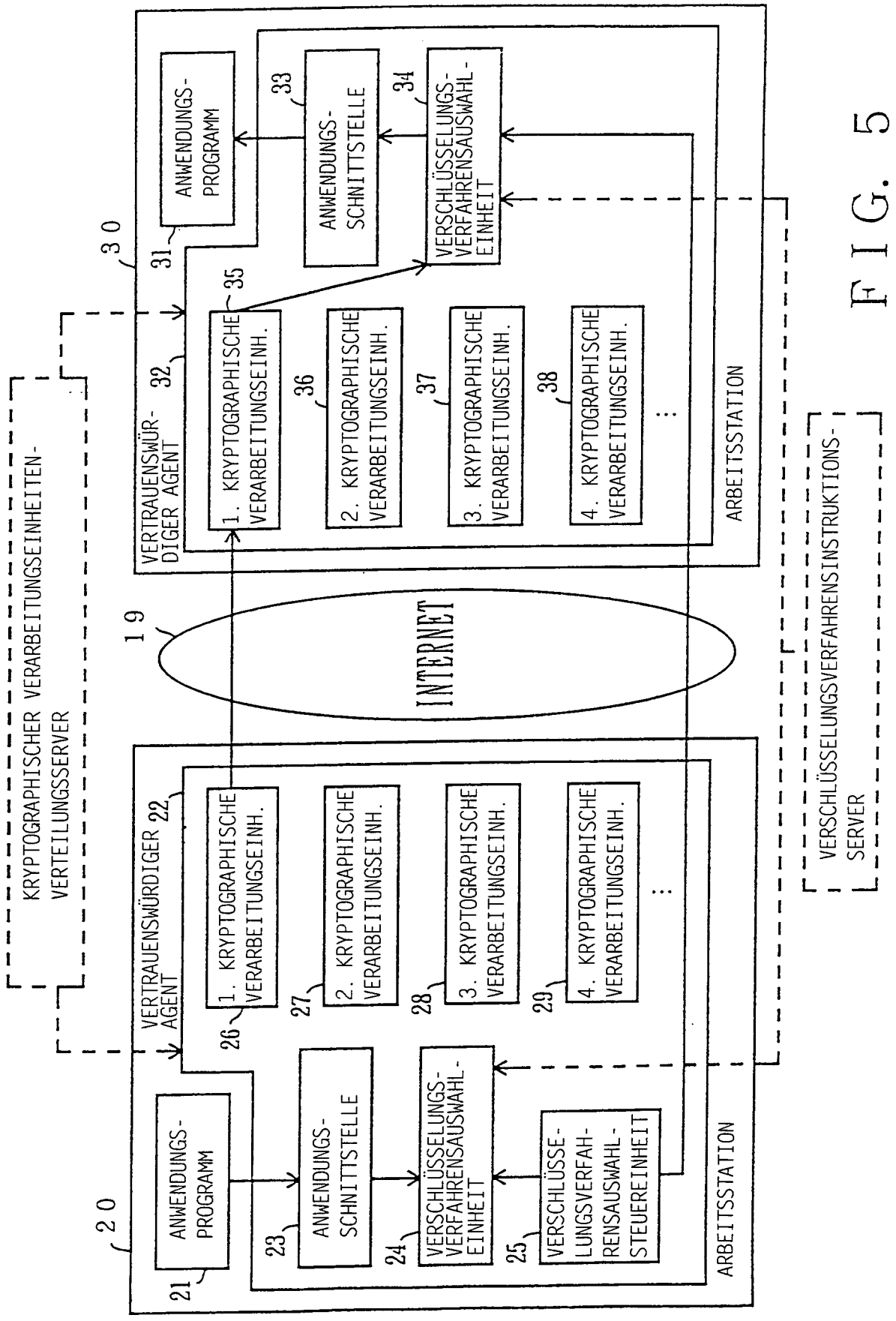

KRYPTOGRAPHISCHER VERARBEITUNGSEINHEITEN-VERTEILUNGSSERVER
VERTRAUENSWÜRDIGER AGENT
32
1. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
36
2. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
37
3. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
38
4. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
ANWENDUNGS-PROGRAMM
31
ANWENDUNGS-SCHNITTSTELLE
33
VERSCHLÜSSELUNGS-VERFAHRENSAUSWAHL-EINHEIT
34
35
ARBEITSSTATION
30
INTERNET
19
VERTRAUENSWÜRDIGER AGENT
22
1. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
26
2. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
27
3. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
28
4. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
29
ANWENDUNGS-PROGRAMM
21
ANWENDUNGS-SCHNITTSTELLE
23
VERSCHLÜSSELUNGS-VERFAHRENSAUSWAHL-EINHEIT
24
VERSCHLÜSSE-LUNGSVERFAH-RENSAUSWAHL-STEUEREINHEIT
25
ARBEITSSTATION
20
VERSCHLÜSSELUNGSVERFAHRENSINSTRUKTIONS-SERVER

FIG. 5

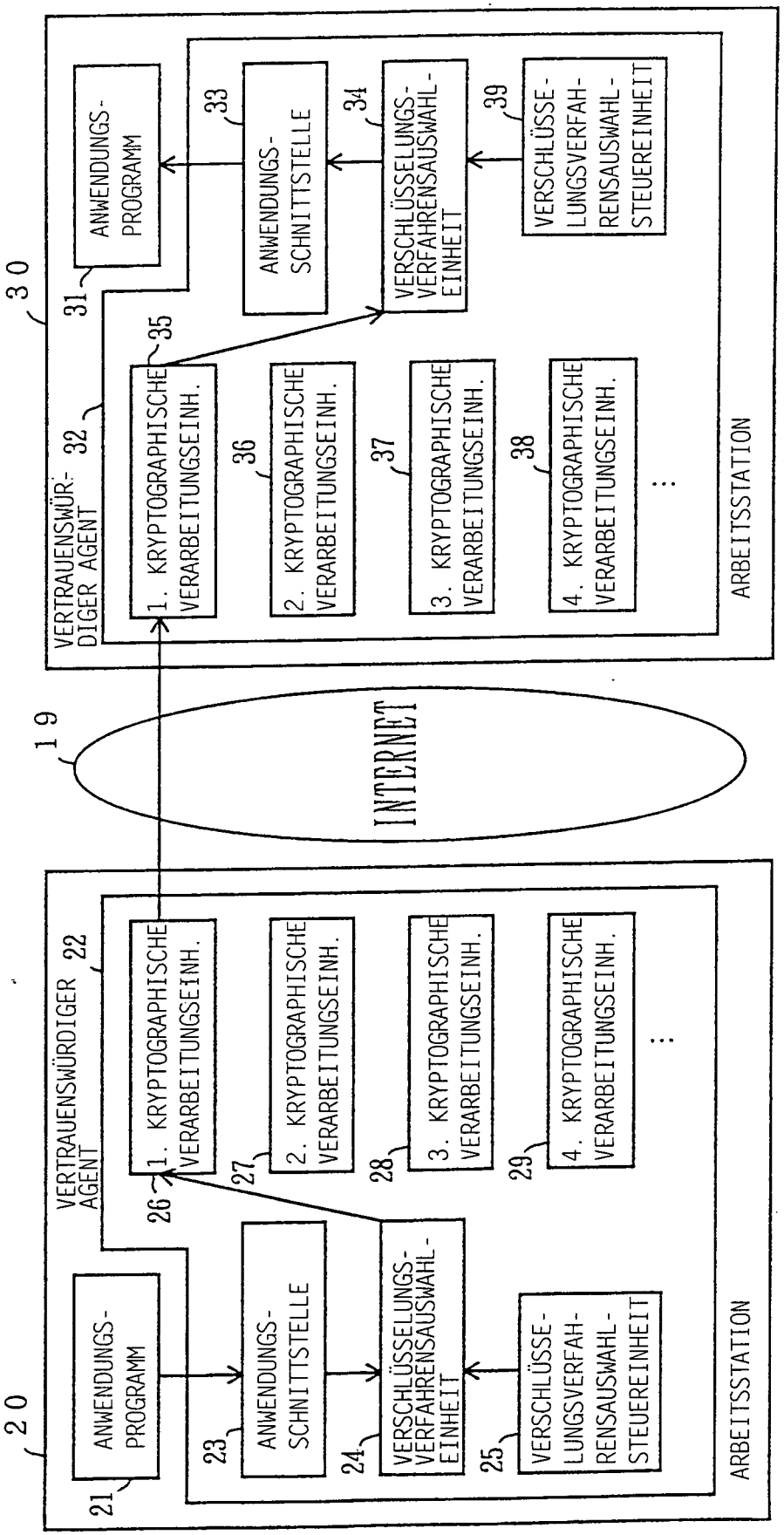

20
ARBEITSSTATION
VERTRAUENSWÜRDIGER AGENT
22
ANWENDUNGS-PROGRAMM
21
ANWENDUNGS-SCHNITTSTELLE
23
VERSCHLÜSSELUNGS-VERFAHRENSAUSWAHL-EINHEIT
24
VERSCHLÜSSELUNGSVERFAHRENSAUSWAHL-STEUEREINHEIT
25
1. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
26
2. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
27
3. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
28
4. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
29
19
INTERNET
30
ARBEITSSTATION
VERTRAUENSWÜRDIGER AGENT
32
ANWENDUNGS-PROGRAMM
31
ANWENDUNGS-SCHNITTSTELLE
33
VERSCHLÜSSELUNGS-VERFAHRENSAUSWAHL-EINHEIT
34
VERSCHLÜSSELUNGSVERFAHRENSAUSWAHL-STEUEREINHEIT
39
1. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
35
2. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
36
3. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
37
4. KRYPTOGRAPHISCHE VERARBEITUNGSEINH.
38

FIG. 6

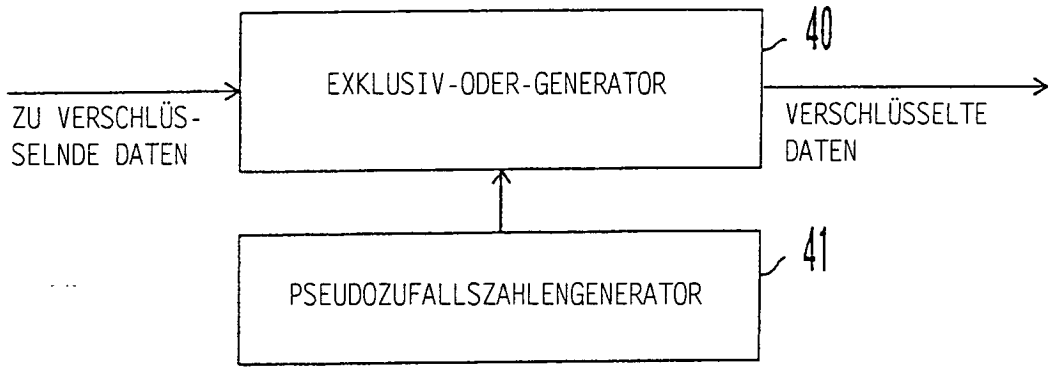

40
EXKLUSIV-ODER-GENERATOR
ZU VERSCHLÜS-
SELNDE DATEN
VERSCHLÜSSELTE
DATEN
41
PSEUDOZUFALLSZAHLENGENERATOR

FIG. 7A

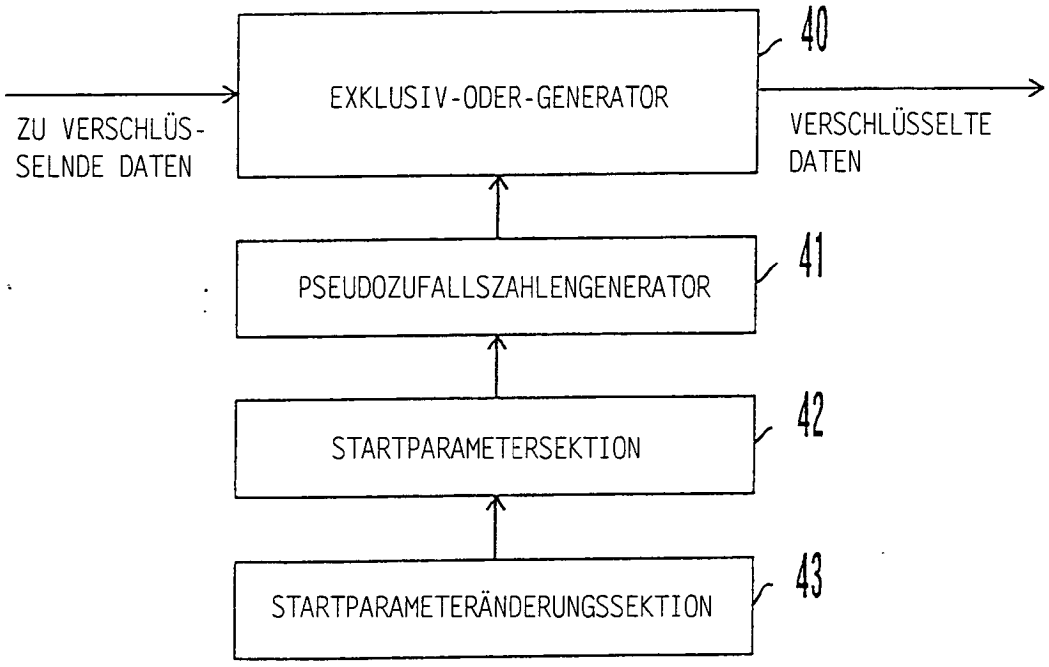

40
EXKLUSIV-ODER-GENERATOR
ZU VERSCHLÜS-
SELNDE DATEN
VERSCHLÜSSELTE
DATEN
41
PSEUDOZUFALLSZAHLENGENERATOR
42
STARTPARAMETERSEKTION
43
STARTPARAMETERÄNDERUNGSSEKTION

FIG. 7B

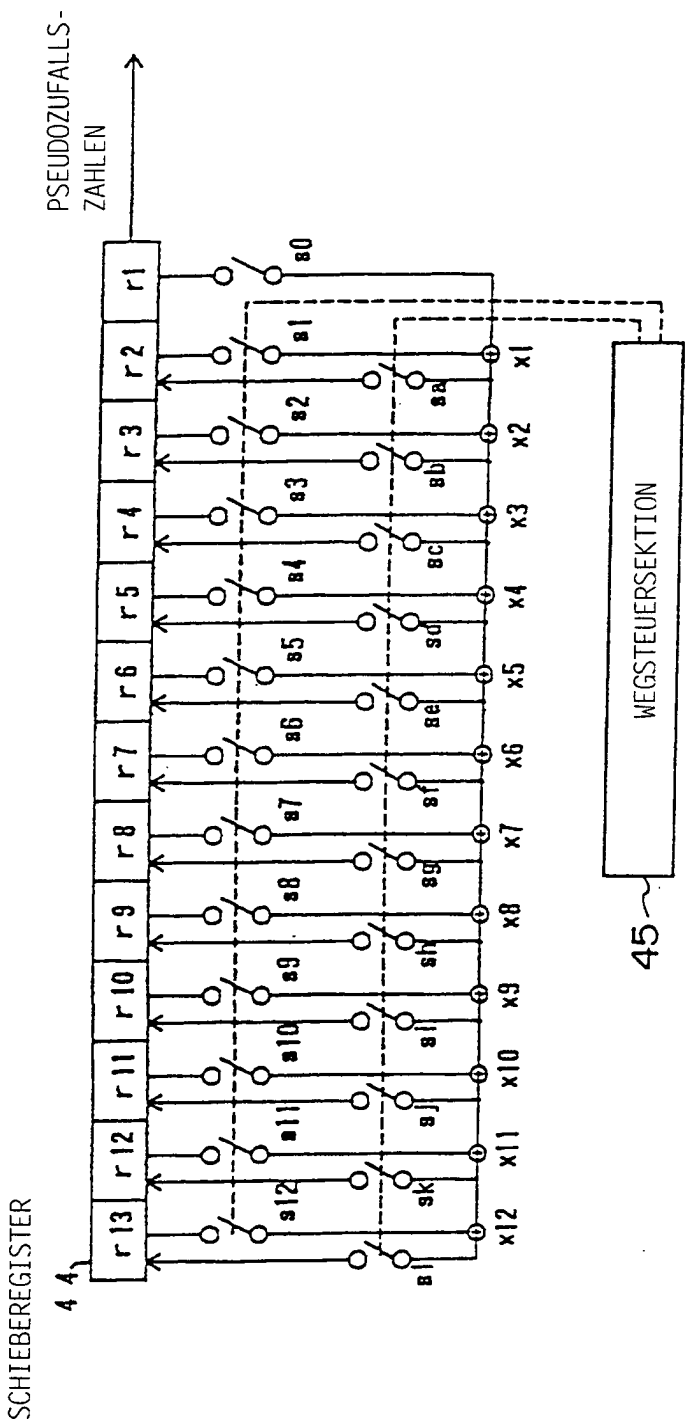

PSEUDOZUFALLS-
ZAHLEN
SCHIEBEREGISTER
44
r1
r2
r3
r4
r5
r6
r7
r8
r9
r10
r11
r12
r13
s0
s1
s2
s3
s4
s5
s6
s7
s8
s9
s10
s11
s12
sa
sb
sc
sd
se
sf
sg
sh
si
sj
sk
sl
x1
x2
x3
x4
x5
x6
x7
x8
x9
x10
x11
x12
WEGSTEUERSEKTION
45

FIG. 8

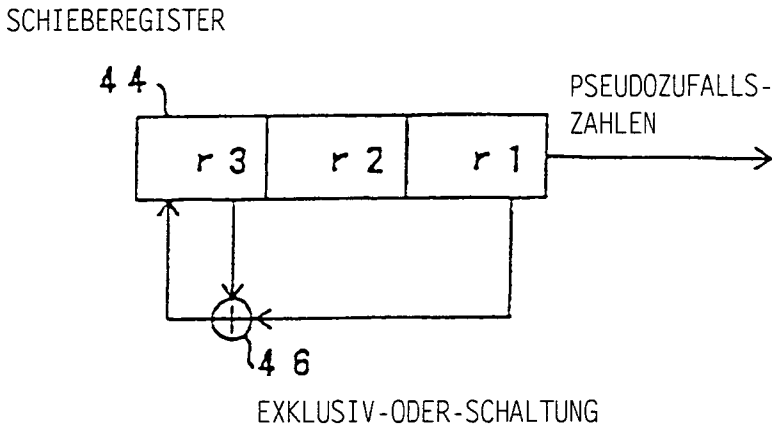

SCHIEBEREGISTER
44
PSEUDOZUFALLS-
ZAHLEN
r 3
r 2
r 1
46
EXKLUSIV-ODER-SCHALTUNG

FIG. 9

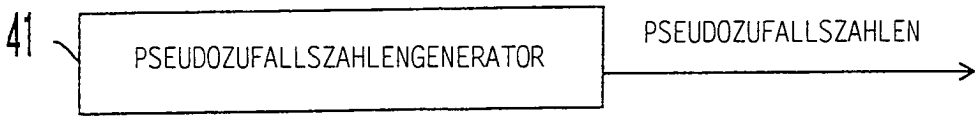


FIG. 10A

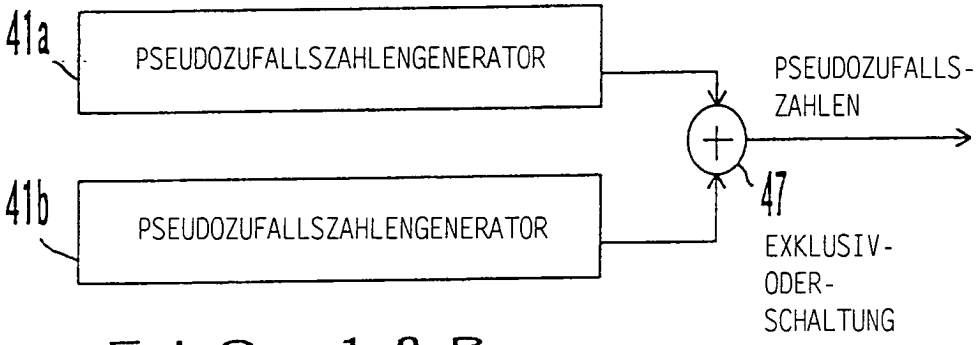


FIG. 10B

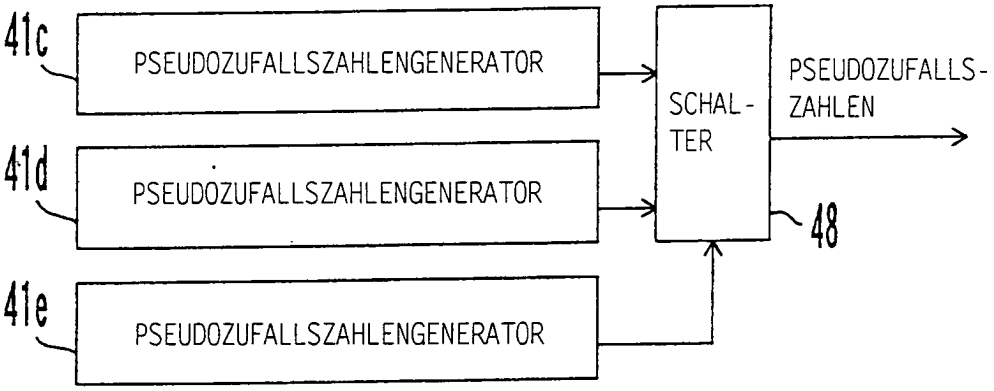


FIG. 10C

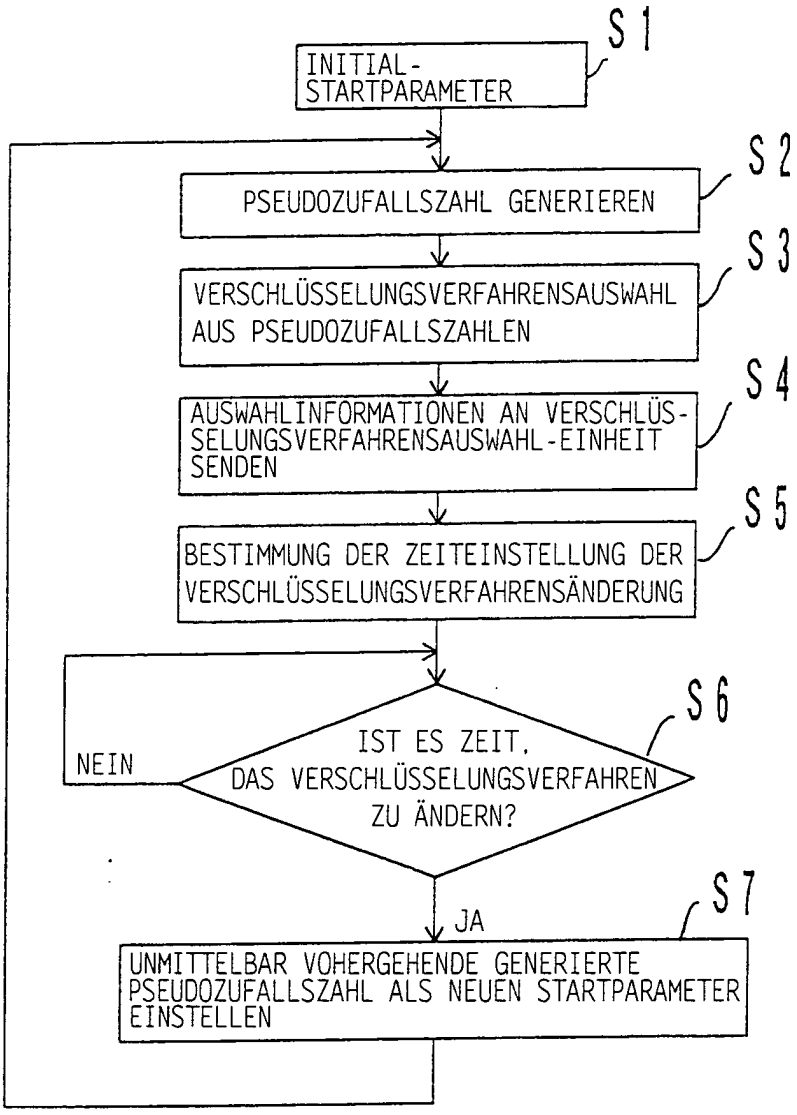

S1
INITIAL-STARTPARAMETER
S2
PSEUDOZUFALLSZAHL GENERIEREN
S3
VERSCHLÜSSELUNGSVERFAHRENSAUSWAHL AUS PSEUDOZUFALLSZAHLEN
S4
AUSWAHLINFORMATIONEN AN VERSCHLÜSSELUNGSVERFAHRENSAUSWAHL-EINHEIT SENDEN
S5
BESTIMMUNG DER ZEITEINSTELLUNG DER VERSCHLÜSSELUNGSVERFAHRENSÄNDERUNG
S6
IST ES ZEIT, DAS VERSCHLÜSSELUNGSVERFAHREN ZU ÄNDERN?
NEIN
JA
S7
UNMITTELBAR VOHERGEHENDE GENERIERTE PSEUDOZUFALLSZAHL ALS NEUEN STARTPARAMETER EINSTELLEN

FIG. 11

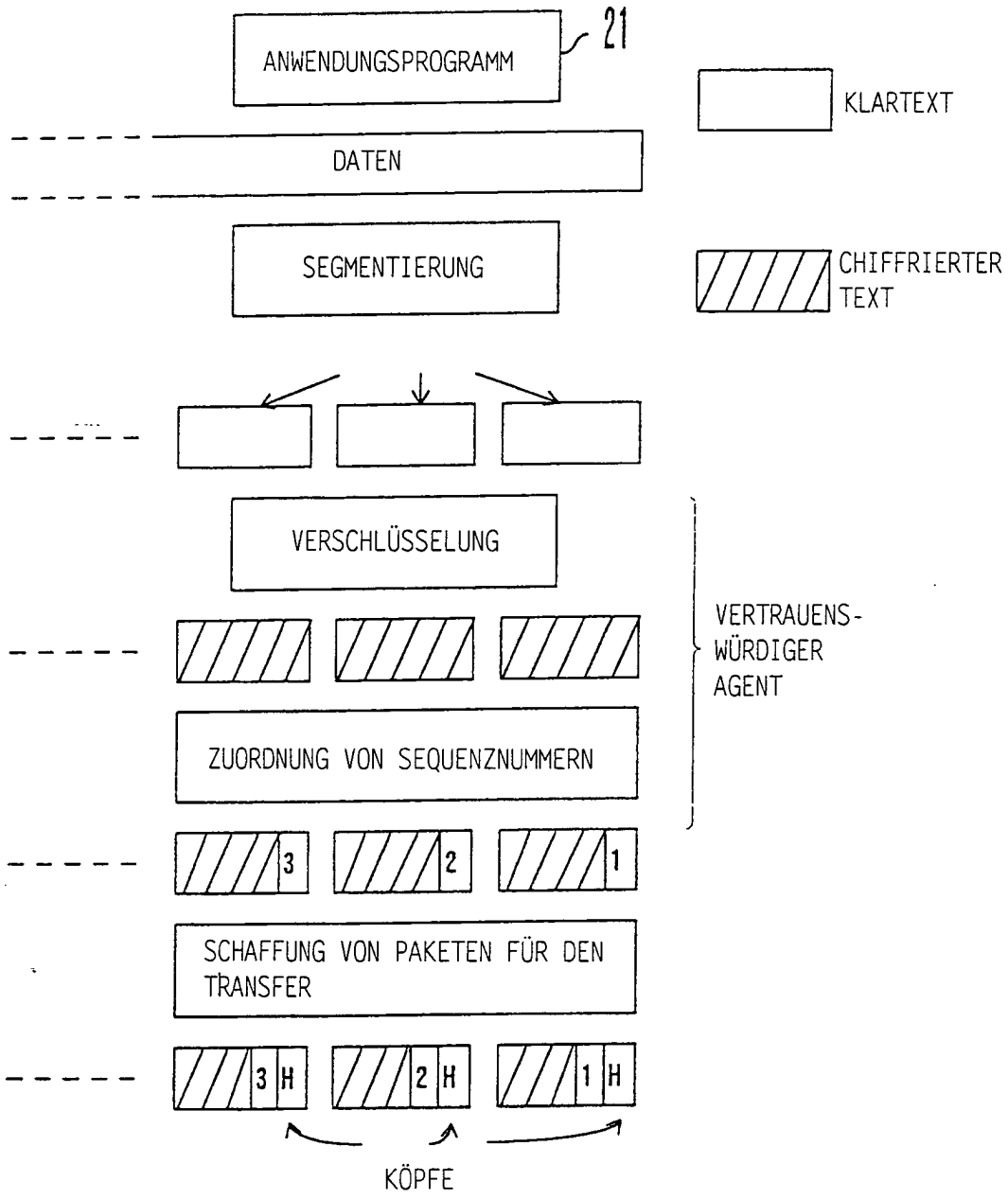


FIG. 12A

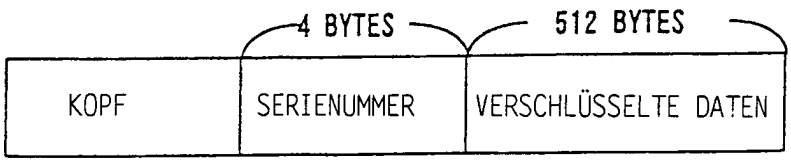


FIG. 12B

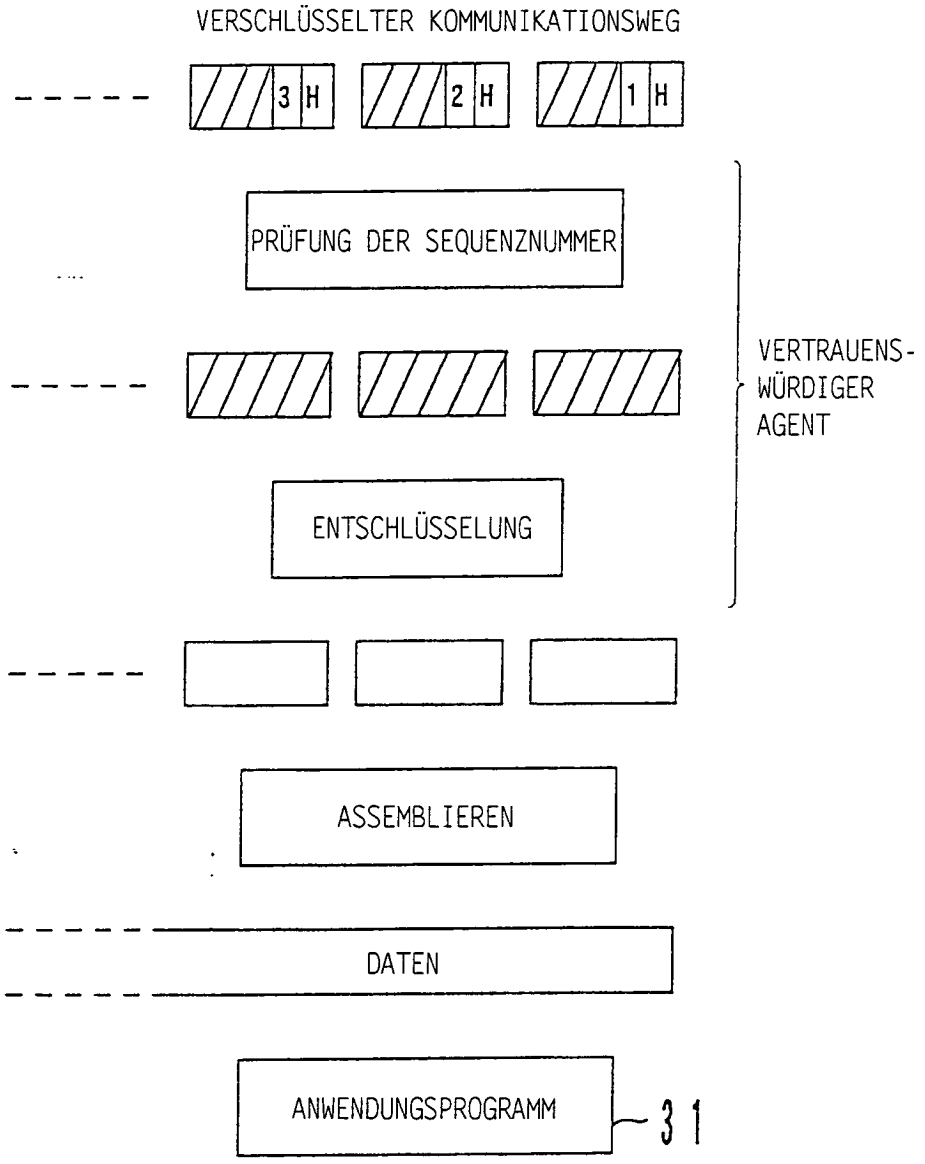

VERSCHLÜSSELTER KOMMUNIKATIONSWEG
3 H
2 H
1 H
PRÜFUNG DER SEQUENZNUMMER
VERTRAUENS-
WÜRDIGER
AGENT
ENTSCHLÜSSELUNG
ASSEMBLIEREN
DATEN
ANWENDUNGSPROGRAMM
31

FIG. 13

F I G. 1 4 A

F I G. 1 4 B

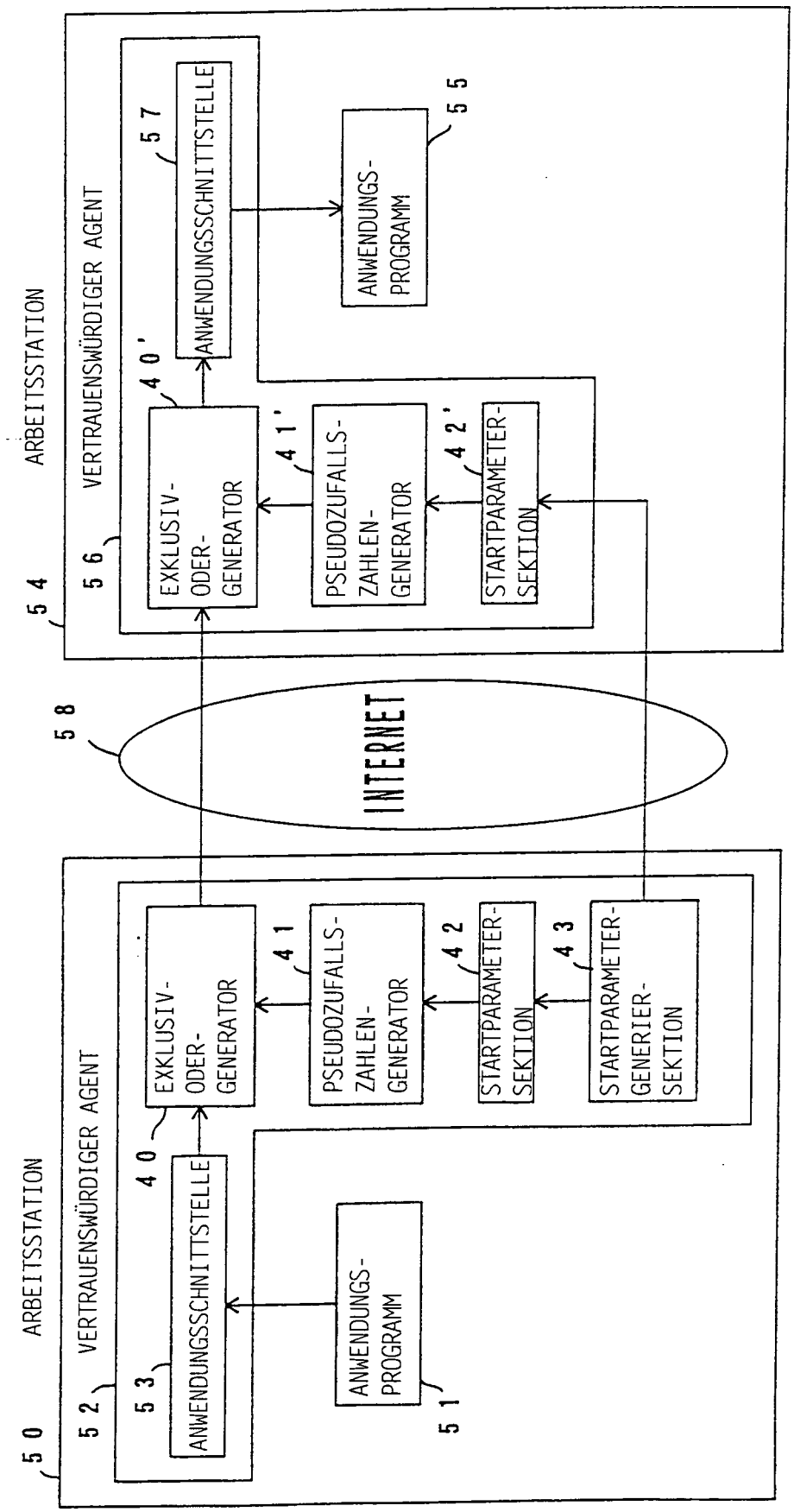

50
ARBEITSSTATION
52
VERTRAUENSWÜRDIGER AGENT
53
ANWENDUNGSSCHNITTSTELLE
40
EXKLUSIV-ODER-GENERATOR
41
PSEUDOZUFALLS-ZAHLEN-GENERATOR
42
STARTPARAMETER-SEKTION
43
STARTPARAMETER-GENERIER-SEKTION
51
ANWENDUNGS-PROGRAMM
58
INTERNET
54
ARBEITSSTATION
56
VERTRAUENSWÜRDIGER AGENT
40'
EXKLUSIV-ODER-GENERATOR
41'
PSEUDOZUFALLS-ZAHLEN-GENERATOR
42'
STARTPARAMETER-SEKTION
57
ANWENDUNGSSCHNITTSTELLE
55
ANWENDUNGS-PROGRAMM

FIG. 15

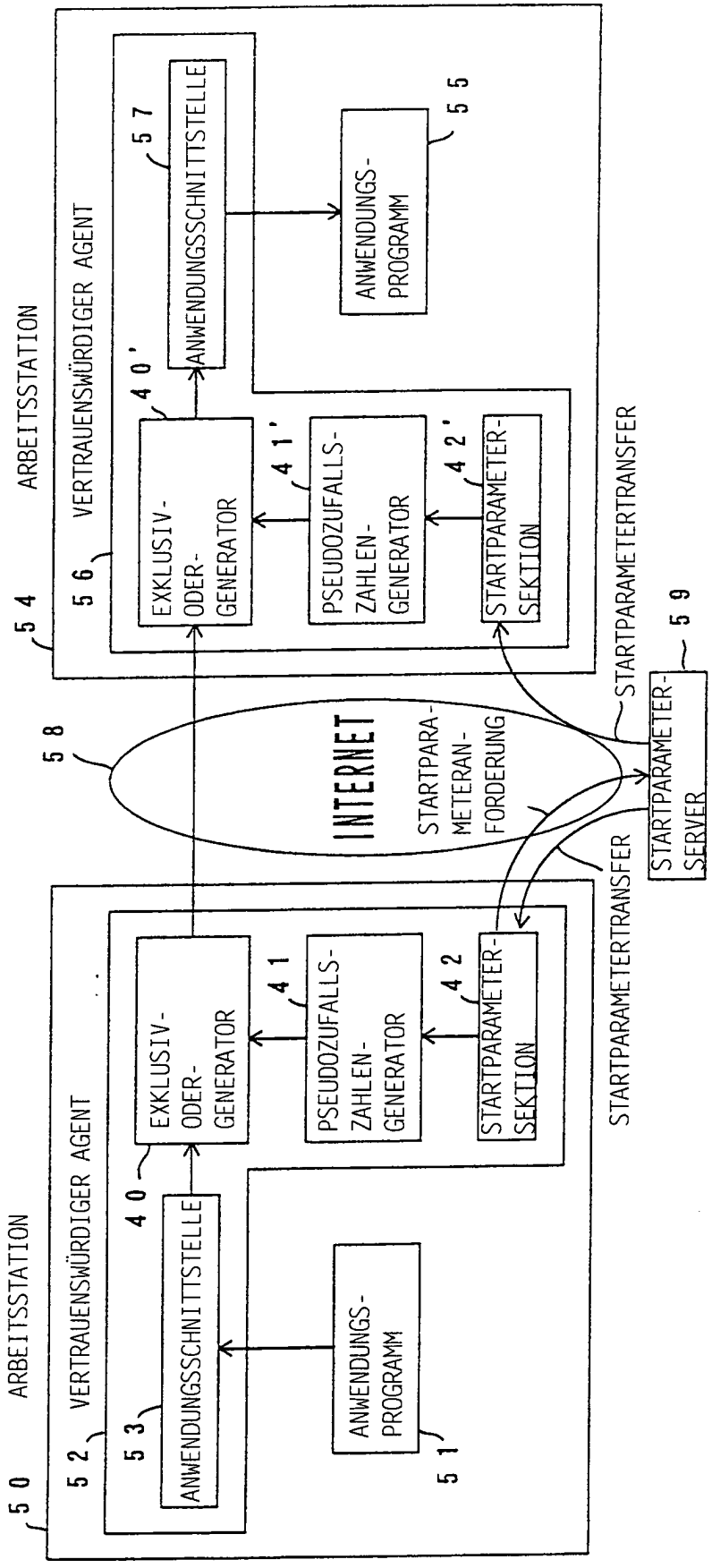

ARBEITSSTATION
VERTRAUENSWÜRDIGER AGENT
ANWENDUNGSSCHNITTSTELLE
EXKLUSIV-ODER-GENERATOR
PSEUDOZUFALLS-ZAHLEN-GENERATOR
STARTPARAMETER-SEKTION
ANWENDUNGS-PROGRAMM
INTERNET
STARTPARA-METERAN-FORDERUNG
STARTPARAMETERTRANSFER
STARTPARAMETER-SERVER
50
51
52
53
54
55
56
57
58
59
40
41
42
40'
41'
42'

FIG. 16

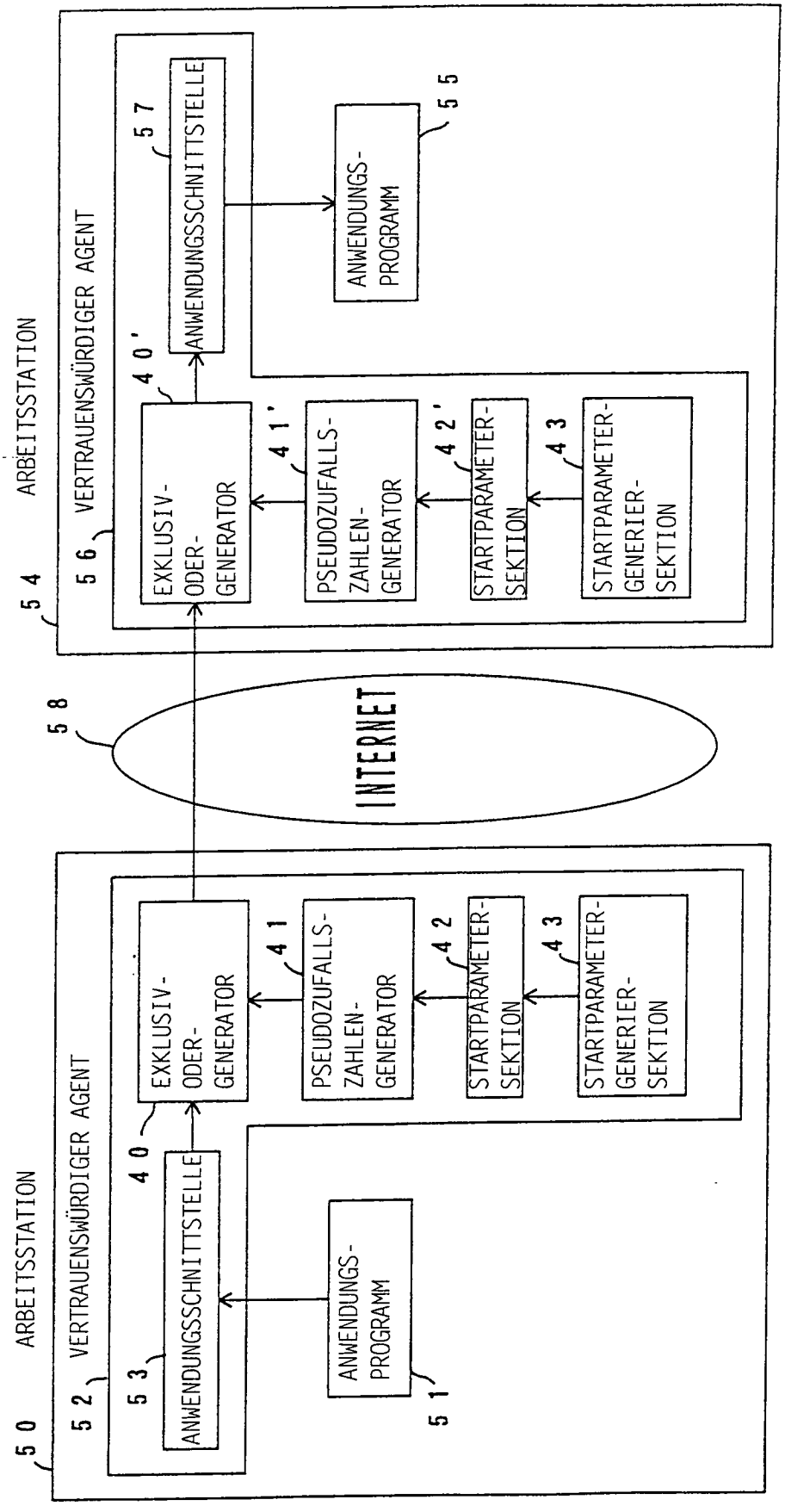

50
ARBEITSSTATION
52
VERTRAUENSWÜRDIGER AGENT
53
ANWENDUNGSSCHNITTSTELLE
40
EXKLUSIV-ODER-GENERATOR
41
PSEUDOZUFALLS-ZAHLEN-GENERATOR
42
STARTPARAMETER-SEKTION
43
STARTPARAMETER-GENERIER-SEKTION
51
ANWENDUNGS-PROGRAMM
58
INTERNET
54
ARBEITSSTATION
56
VERTRAUENSWÜRDIGER AGENT
40'
EXKLUSIV-ODER-GENERATOR
41'
PSEUDOZUFALLS-ZAHLEN-GENERATOR
42'
STARTPARAMETER-SEKTION
43
STARTPARAMETER-GENERIER-SEKTION
57
ANWENDUNGSSCHNITTSTELLE
55
ANWENDUNGS-PROGRAMM

FIG. 17

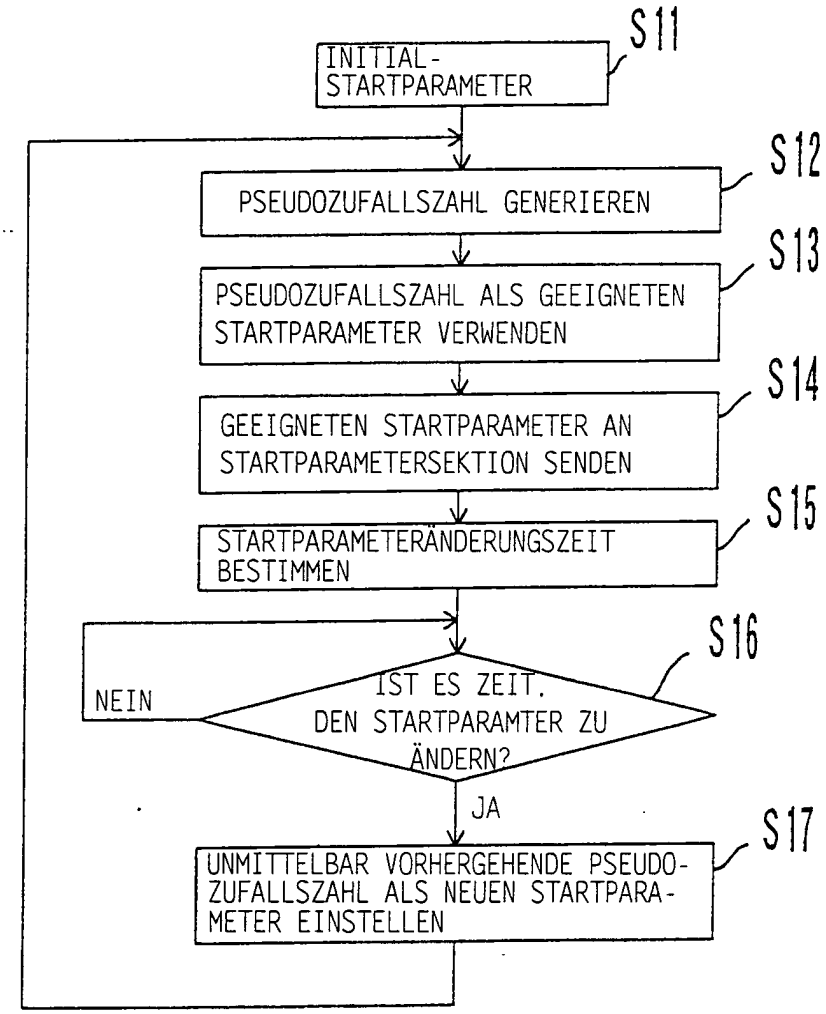

S11
INITIAL-STARTPARAMETER
S12
PSEUDOZUFALLSZAHL GENERIEREN
S13
PSEUDOZUFALLSZAHL ALS GEEIGNETEN STARTPARAMETER VERWENDEN
S14
GEEIGNETEN STARTPARAMETER AN STARTPARAMETERSEKTION SENDEN
S15
STARTPARAMETERÄNDERUNGSZEIT BESTIMMEN
S16
IST ES ZEIT, DEN STARTPARAMTER ZU ÄNDERN?
NEIN
JA
S17
UNMITTELBAR VORHERGEHENDE PSEUDO-ZUFALLSZAHL ALS NEUEN STARTPARA-METER EINSTELLEN

FIG. 18

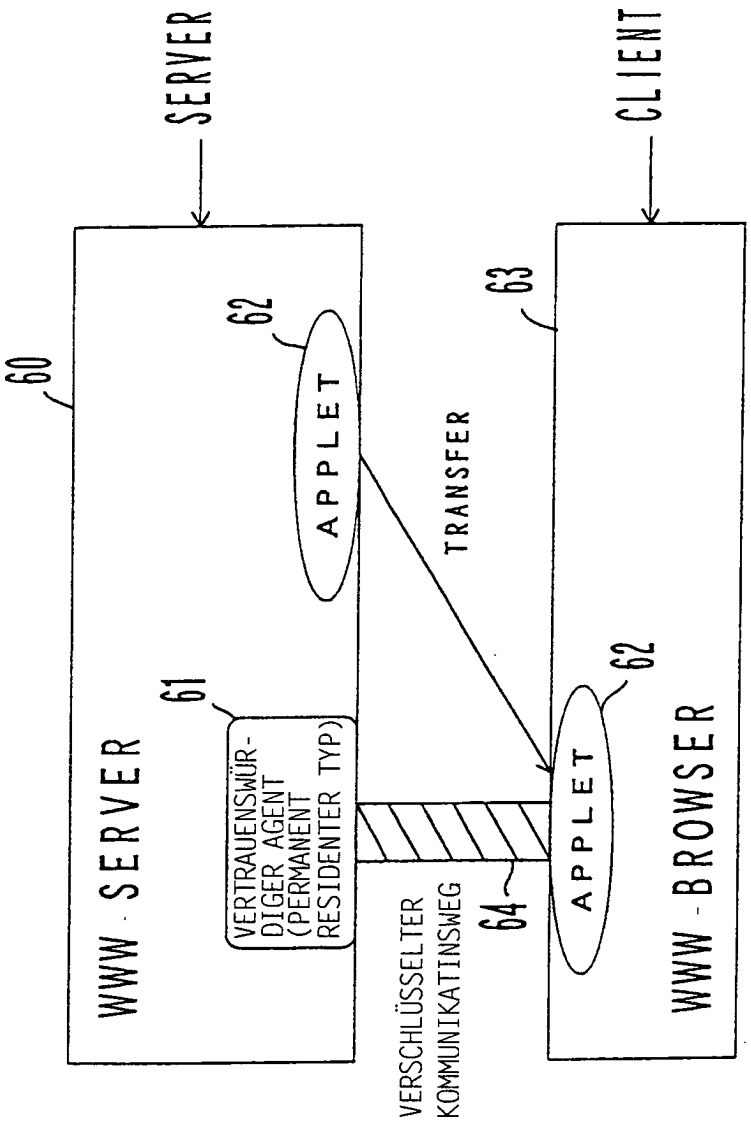

SERVER
CLIENT
60
62
APPLET
63
TRANSFER
61
VERTRAUENSWÜR-
DIGER AGENT
(PERMANENT
RESIDENTER TYP)
62
APPLET
WWW - SERVER
WWW - BROWSER
VERSCHLÜSSELTER
KOMMUNIKATINSWEG
64

FIG. 19

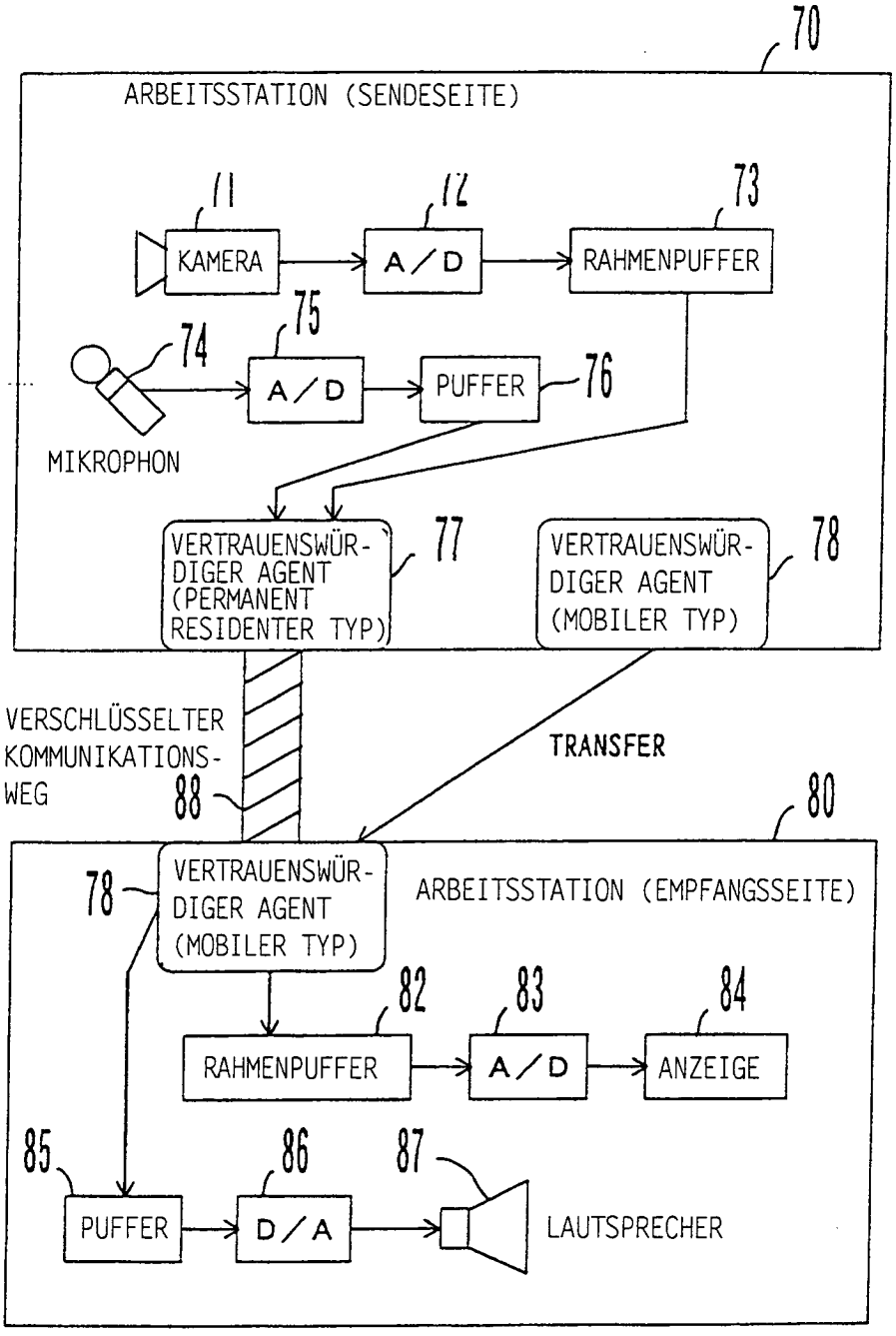

70
ARBEITSSTATION (SENDESEITE)
11
KAMERA
12
A/D
73
RAHMENPUFFER
74
MIKROPHON
75
A/D
PUFFER
76
VERTRAUENSWÜR-
DIGER AGENT
(PERMANENT
RESIDENTER TYP)
77
VERTRAUENSWÜR-
DIGER AGENT
(MOBILER TYP)
78
VERSCHLÜSSELTER
KOMMUNIKATIONS-
WEG
88
TRANSFER
80
78
VERTRAUENSWÜR-
DIGER AGENT
(MOBILER TYP)
ARBEITSSTATION (EMPFANGSSEITE)
82
RAHMENPUFFER
83
A/D
84
ANZEIGE
85
PUFFER
86
D/A
87
LAUTSPRECHER

FIG. 20

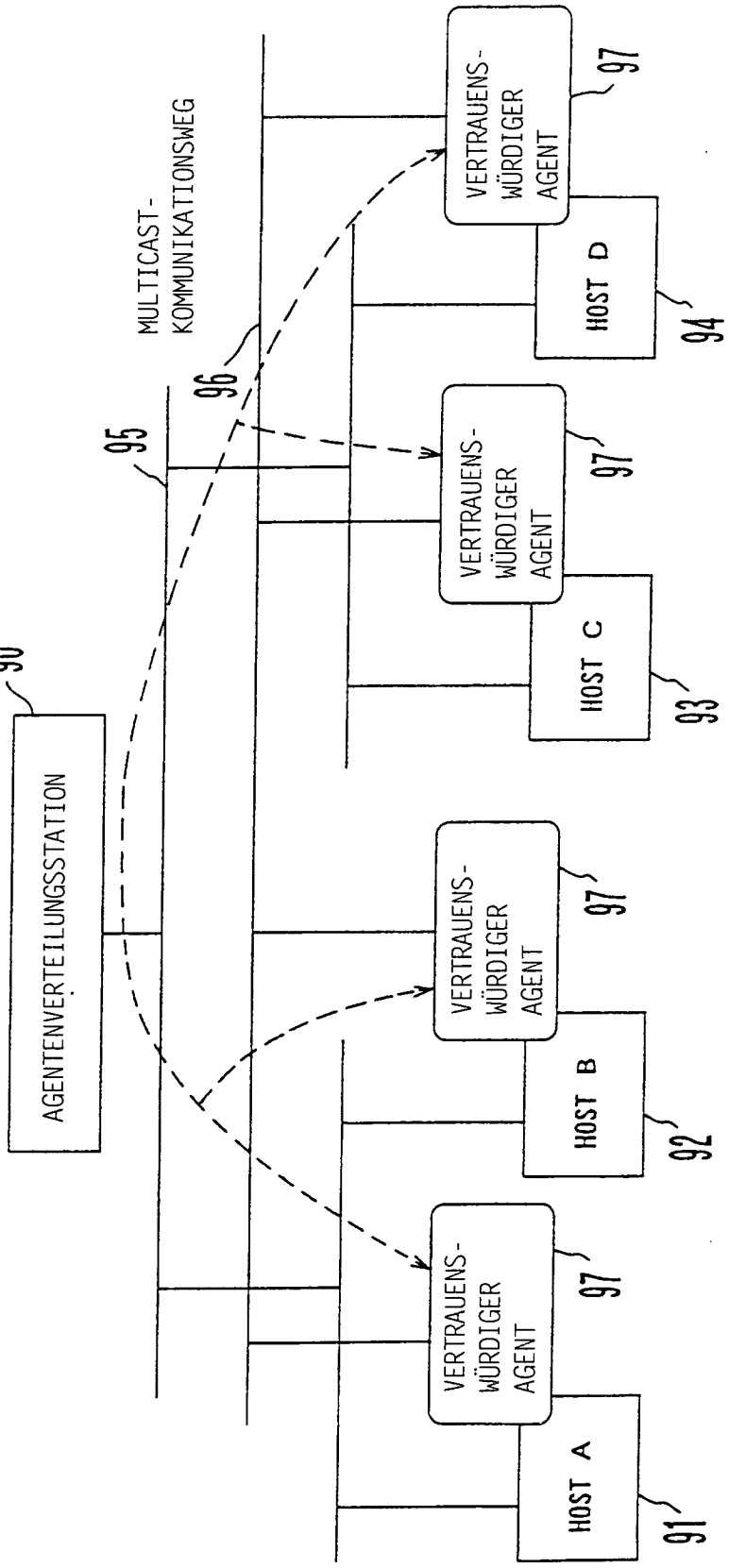

AGENTENVERTEILUNGSSTATION
90
MULTICAST-
KOMMUNIKATIONSWEG
95
96
VERTRAUENS-
WÜRDIGER
AGENT
97
HOST A
91
HOST B
92
HOST C
93
HOST D
94

FIG. 21

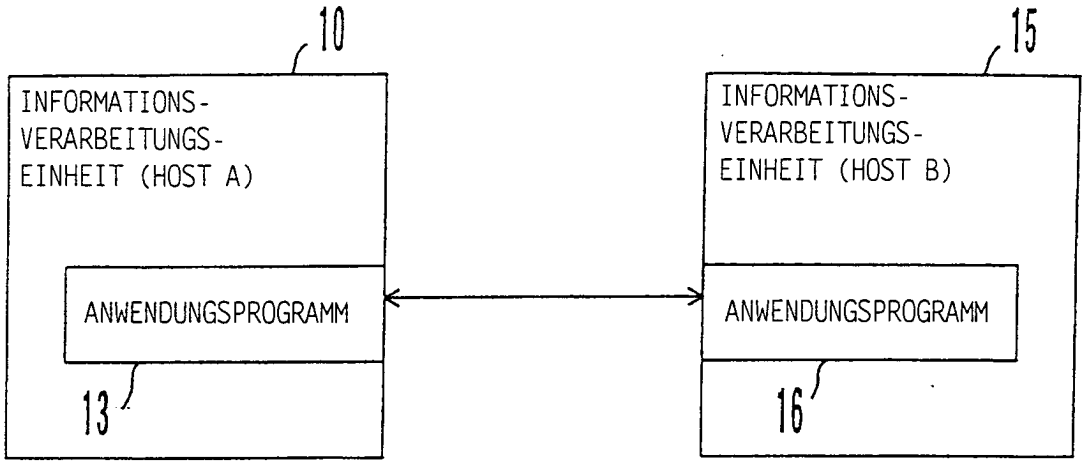


FIG. 22A

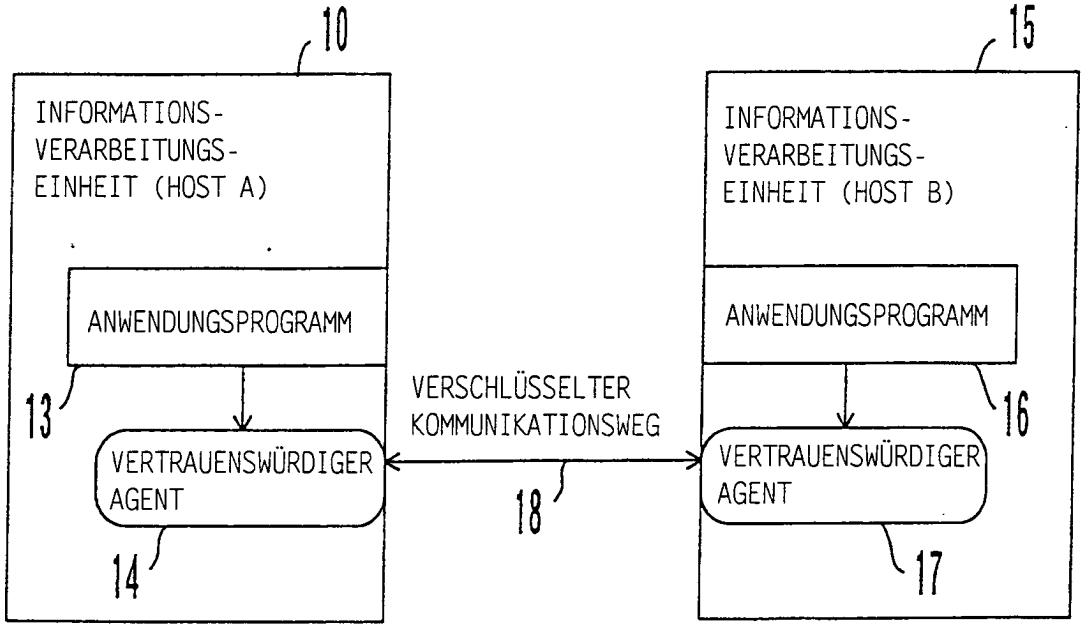


FIG. 22B

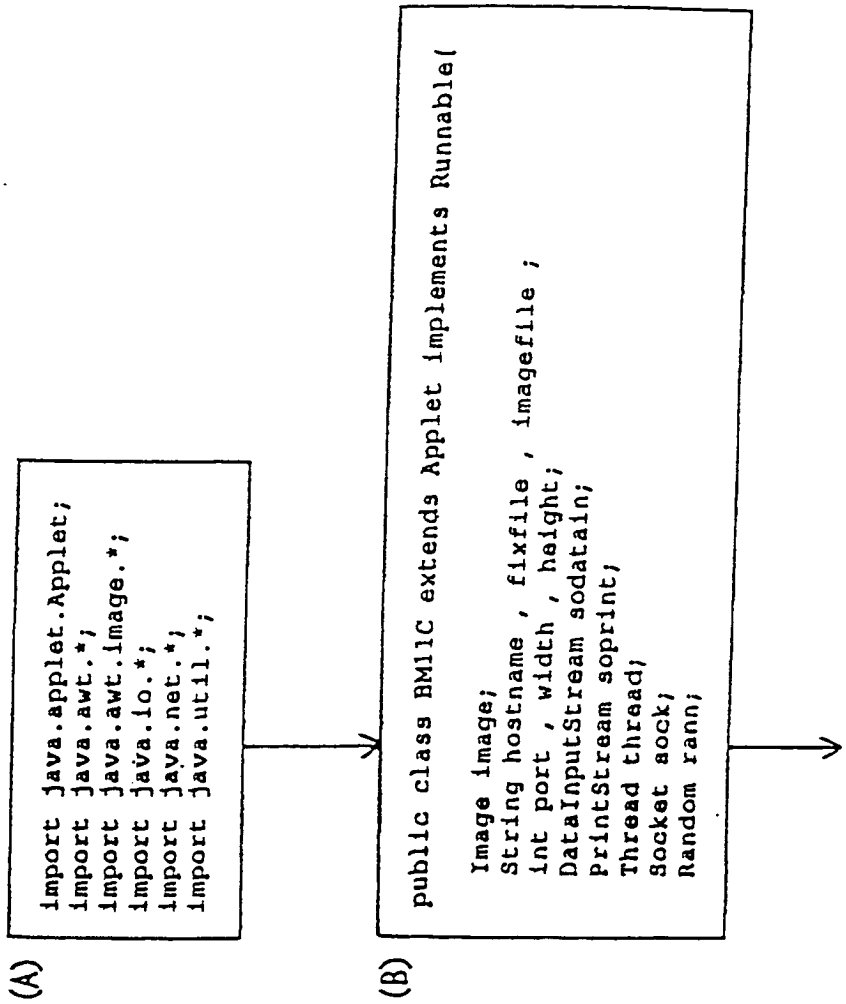

(A)
import java.applet.Applet;
import java.awt.*;
import java.awt.image.*;
import java.io.*;
import java.net.*;
import java.util.*;
(B)
public class BM11C extends Applet implements Runnable{
Image image;
String hostname , fixfile , imagefile ;
int port , width , height;
DataInputStream sodatain;
PrintStream soprint;
Thread thread;
Socket sock;
Random rann;

FIG. 23

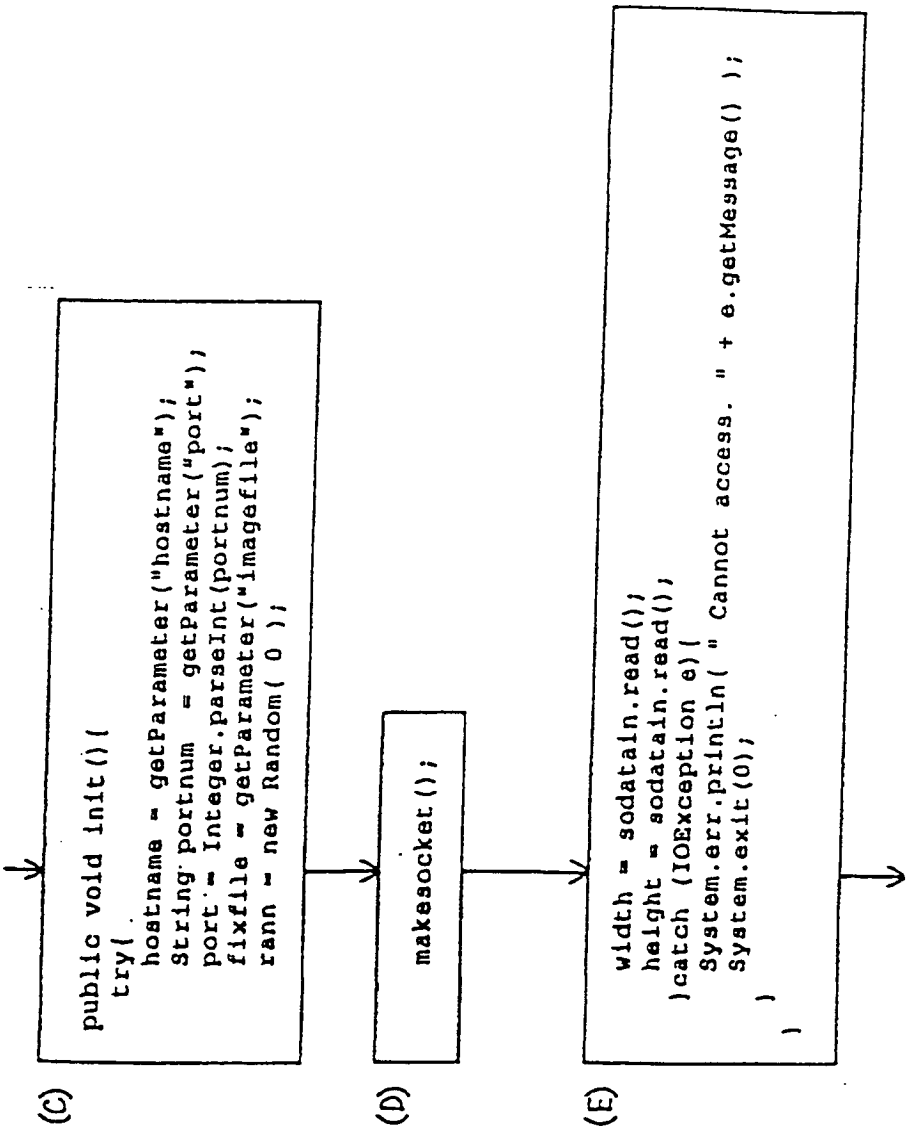

(C)
public void init(){
try{
hostname = getParameter("hostname");
String portnum = getParameter("port");
port = Integer.parseInt(portnum);
fixfile = getParameter("imagefile");
rann = new Random(0);
(D)
makesocket();
(E)
width = sodatain.read();
height = sodatain.read();
}catch (IOException e){
System.err.println(" Cannot access. " + e.getMessage());
System.exit(0);
}
}

FIG. 24

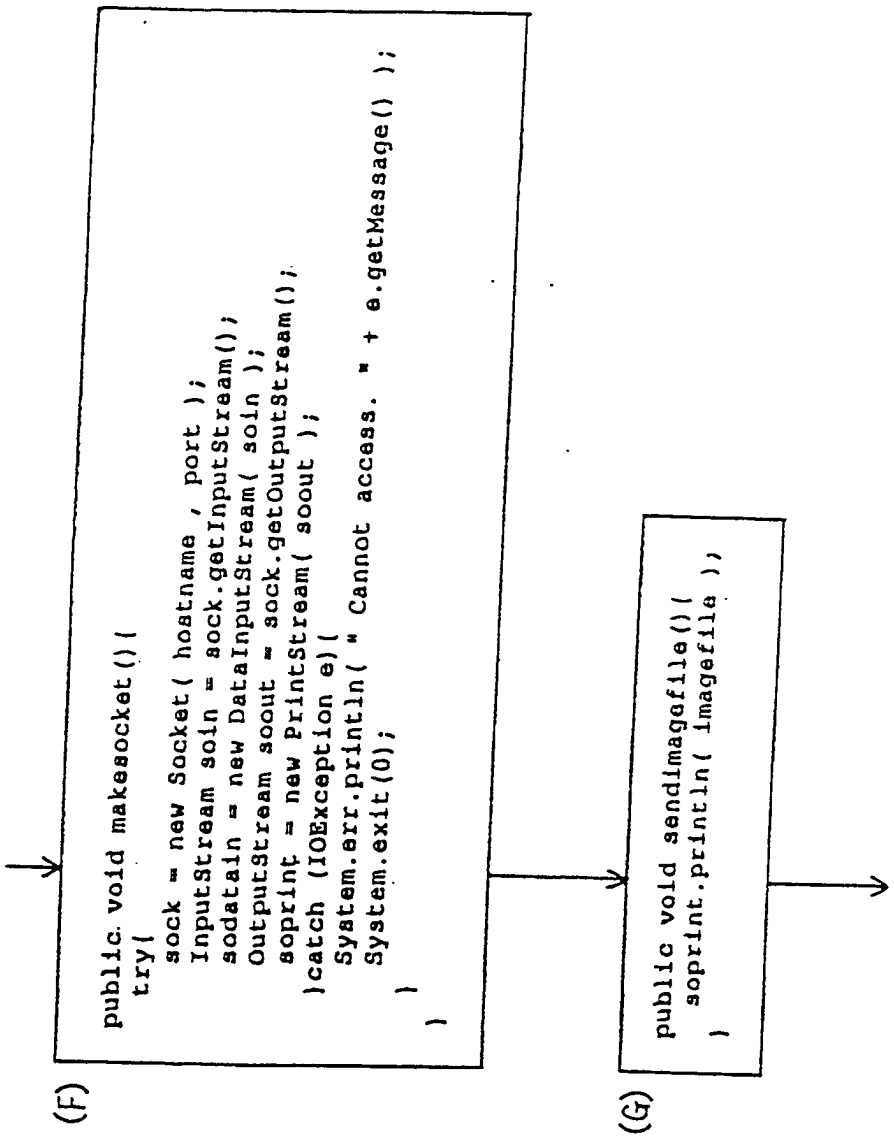

(F)
public void makesocket(){
try{
sock = new Socket(hostname , port);
InputStream soin = sock.getInputStream();
sodatain = new DataInputStream(soin);
OutputStream soout = sock.getOutputStream();
soprint = new PrintStream(soout);
}catch (IOException e){
System.err.println(" Cannot access. " + e.getMessage());
System.exit(0);
}
}
(G)
public void sendimagefile(){
soprint.println(imagefile);
}

FIG. 25

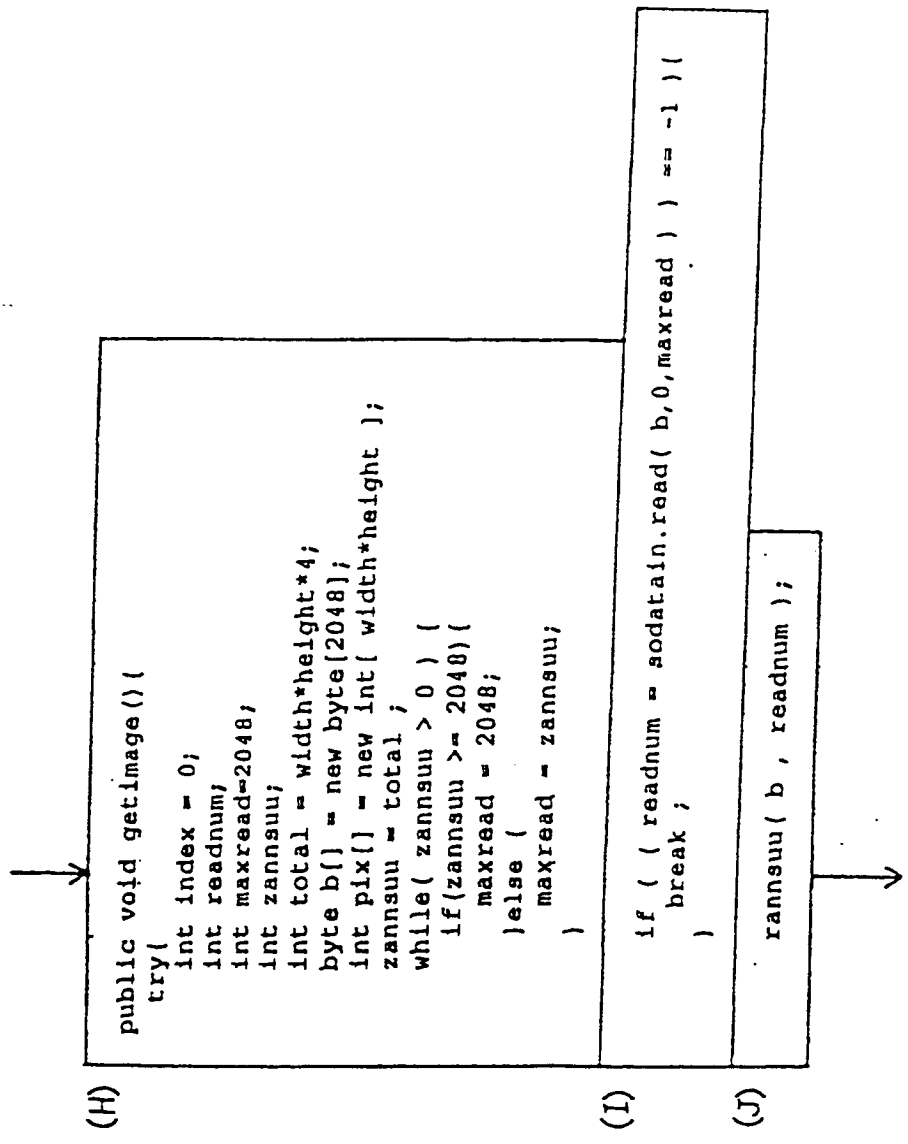

(H)
public void getImage(){
try{
int index = 0;
int readnum;
int maxread=2048;
int zannsuu;
int total = width*height*4;
byte b[] = new byte[2048];
int pix[] = new int(width*height];
zannsuu = total ;
while(zannsuu > 0) {
if(zannsuu >= 2048){
maxread = 2048;
}else {
maxread = zannsuu;
}
(I)
if ((readnum = sodatain.read(b,0,maxread)) == -1){
break ;
}
(J)
rannsuu(b , readnum);

FIG. 26

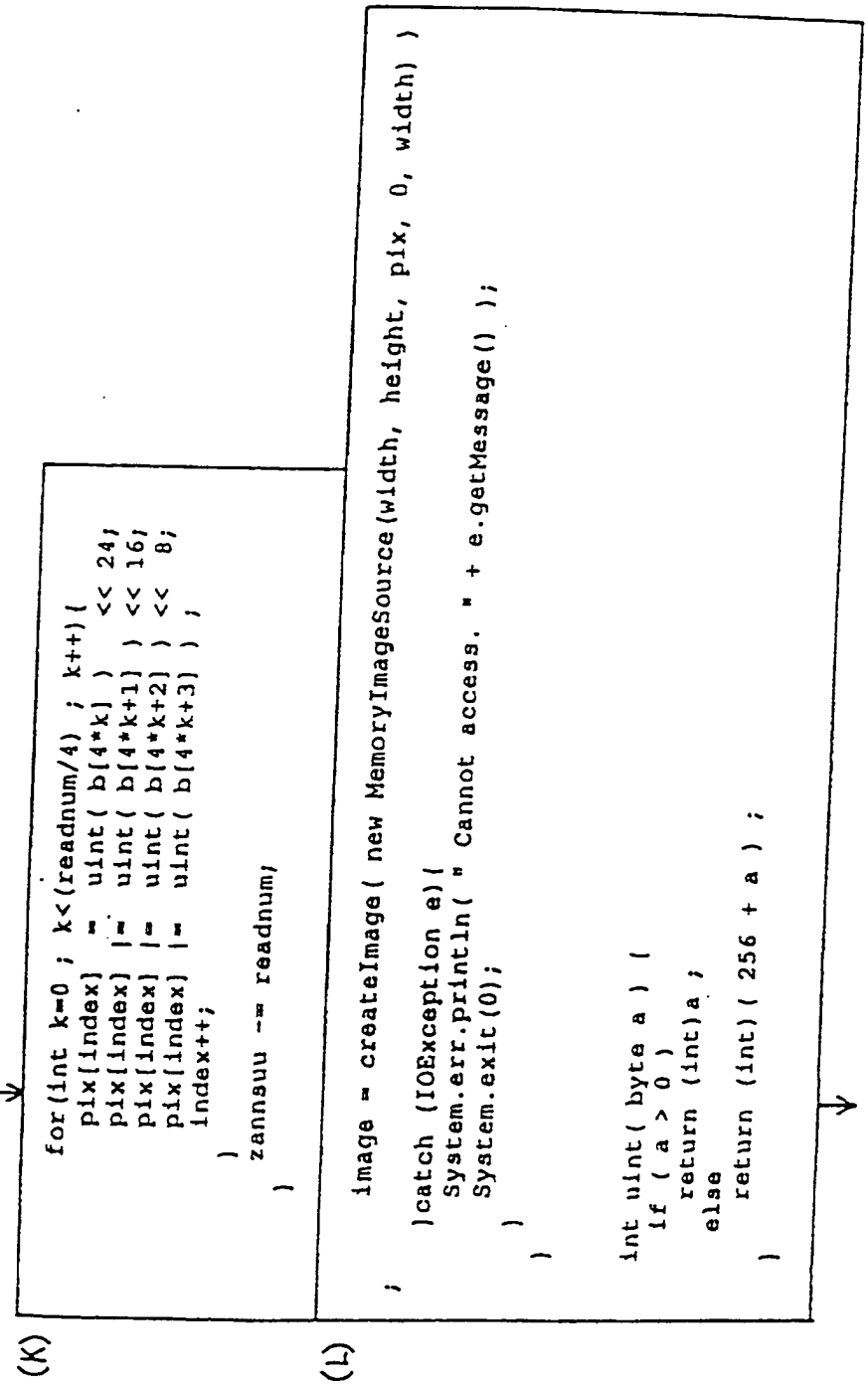

(K)
for(int k=0 ; k<(readnum/4) ; k++){
pix[index] = uint(b[4*k]) << 24;
pix[index] |= uint(b[4*k+1]) << 16;
pix[index] |= uint(b[4*k+2]) << 8;
pix[index] |= uint(b[4*k+3]) ;
index++;
}
zannsuu -= readnum;
}
(L)
image = createImage(new MemoryImageSource(width, height, pix, 0, width))
;
}catch (IOException e){
System.err.println(" Cannot access. " + e.getMessage());
System.exit(0);
}
}
int uint(byte a) {
if (a > 0)
return (int)a ;
else
return (int)(256 + a) ;
}

FIG. 27

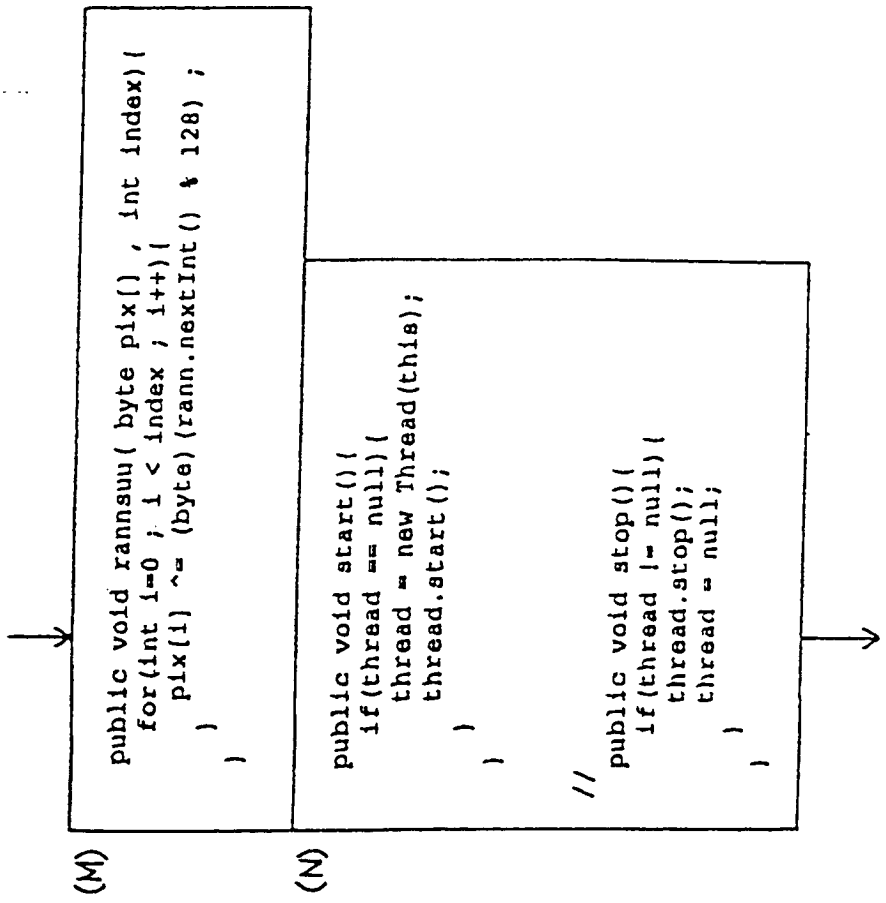

(M)
public void rannsuu(byte pix[] , int index){
for(int i=0 ; i < index ; i++){
pix[i] ^= (byte)(rann.nextInt() % 128) ;
}
}
(N)
public void start(){
if(thread == null){
thread = new Thread(this);
thread.start();
}
}
//
public void stop(){
if(thread != null){
thread.stop();
thread = null;
}
}

FIG. 28

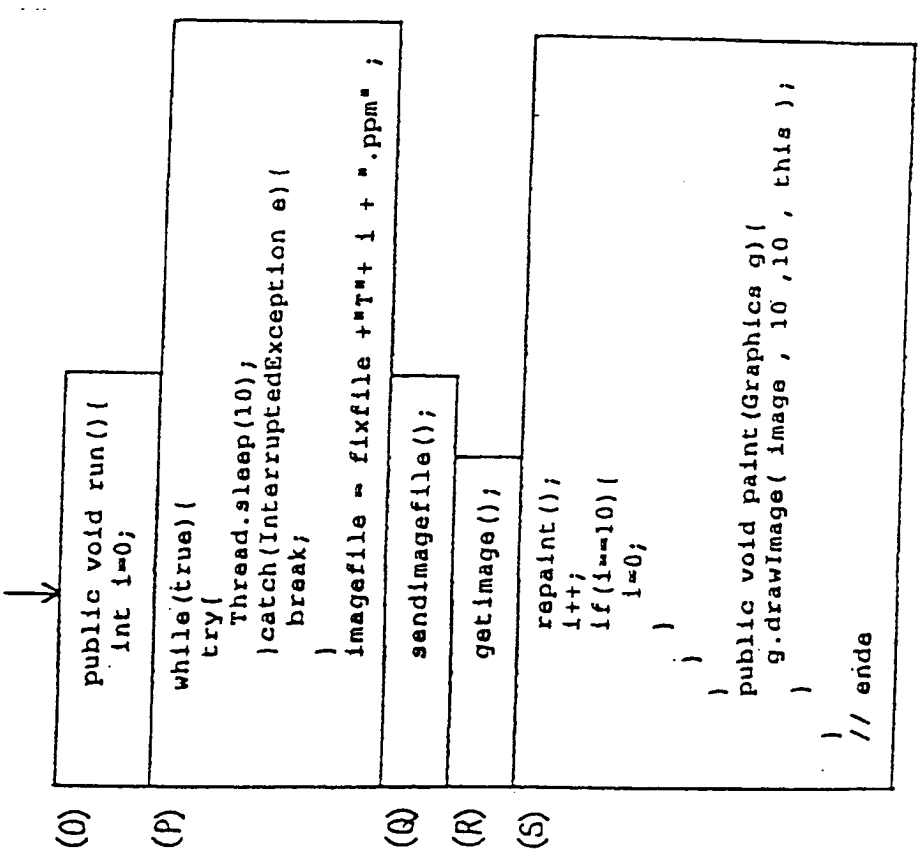

(O)
public void run(){
int i=0;
(P)
while(true){
try{
Thread.sleep(10);
}catch(InterruptedException e){
break;
}
imagefile = fixfile +"T"+ i + ".ppm" ;
(Q)
sendimagefile();
(R)
getImage();
(S)
repaint();
i++;
if(i==10){
i=0;
}
}
}
public void paint(Graphics g){
g.drawImage(image , 10 ,10 , this);
}
},
// ende

FIG. 29