(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3842100号
(P3842100)
(45)発行日　平成18年11月8日(2006.11.8)　(24)登録日　平成18年8月18日(2006.8.18)

(51)Int.Cl.　　F I
H04L 9/32 (2006.01)　　H04L 9/00 675B
G06F 21/20 (2006.01)　　G06F 15/00 330A

請求項の数 5　(全 19 頁)

(21)出願番号　特願2001-316575(P2001-316575)
(22)出願日　平成13年10月15日(2001.10.15)
(65)公開番号　特開2003-124926(P2003-124926A)
(43)公開日　平成15年4月25日(2003.4.25)
審査請求日　平成15年11月14日(2003.11.14)

(73)特許権者　000005108
株式会社日立製作所
東京都千代田区丸の内一丁目６番６号
(74)代理人　110000350
特許業務法人　日東国際特許事務所
(74)代理人　100068504
弁理士　小川　勝男
(74)代理人　100086656
弁理士　田中　恭助
(74)代理人　100094352
弁理士　佐々木　孝
(72)発明者　西原　啓輔
神奈川県横浜市戸塚区戸塚町５０３０番地
株式会社日立製作所　ソフトウェア事業部内

最終頁に続く

(54)【発明の名称】暗号化通信システムにおける認証処理方法及びそのシステム

(57)【特許請求の範囲】

【請求項１】

第１の計算機、前記第１の計算機に対してセション確立要求を行う第２の計算機および前記第１の計算機と前記第２の計算機との間に介在し両計算機間で送受信されるアプリケーションデータを中継する第３の計算機とを有するシステムであり、共通鍵暗号方式に基づいて前記第１の計算機と前記第２の計算機の間で前記アプリケーションデータを暗号化して送受信する暗号化通信システムの認証処理方法において、

前記第２の計算機からの当該セション固有のセションデータを含むセション確立要求に対し、前記第３の計算機によって生成された当該セション固有のセションデータを付加して前記第１の計算機へセション確立要求を送信し、前記第１の計算機によって生成された当該セション固有のセションデータと前記第２の計算機および前記第３の計算機から得られた前記セションデータとを含めた第１の電子証明書を前記第３の計算機へ送信し、前記第３の計算機によって受信した前記第１の電子証明書を署名検証により認証後に前記第１の電子証明書と前記第１の計算機および前記第２の計算機から得られた前記セションデータとを含めた第２の電子証明書を前記第２の計算機へ送信し、前記第２の計算機によって前記第１の計算機、前記第２の計算機および前記第３の計算機によって生成された当該セション固有のセションデータの少なくとも一部と、前記第１の電子証明書及び前記第２の電子証明書を署名検証により認証することを特徴とする認証処理方法。

【請求項２】

前記第２の計算機から暗号化された前記アプリケーションデータを受信した前記第３の

計算機によって前記アプリケーションデータを復号化してデータチェックを行い、前記第１の計算機から暗号化された前記アプリケーションデータを受信した前記第３の計算機によって前記アプリケーションデータを復号化してデータチェックを行うことを特徴とする請求項１記載の認証処理方法。

【請求項３】

セション確立要求を受信する第１の計算機、前記第１の計算機に対してセション確立要求を行う第２の計算機および前記第１の計算機と前記第２の計算機との間に介在し両計算機間で送受信されるアプリケーションデータを中継する第３の計算機とを有するシステムであり、共通鍵暗号方式に基づいて前記第１の計算機と前記第２の計算機との間で前記アプリケーションデータを暗号化して送受信する暗号化通信システムの認証処理方法において、

前記第２の計算機からの当該セション固有のセションデータを含むセション確立要求に対し、前記第３の計算機によって生成された当該セション固有のセションデータを付加して前記第１の計算機へセション確立要求を送信し、前記第１の計算機によって生成された当該セション固有のセションデータと前記第２の計算機および前記第３の計算機から得られた前記セションデータを前記第３の計算機を介して前記第２の計算機へ送信し、前記第３の計算機によって、前記第１の計算機、前記第２の計算機および前記第３の計算機によって生成された前記セションデータのうちの少なくとも一部を含む第１の電子証明書を前記第２の計算機へ送信し、前記第２の計算機によって受信した前記第１の計算機、前記第２の計算機および前記第３の計算機によって生成された前記セションデータと前記第１の電子証明書を署名検証により認証後に前記第１の計算機から得られた前記セションデータを含む第２の電子証明書を前記第３の計算機へ送信し、前記第３の計算機によって受信した前記第２の電子証明書を署名検証により認証後に前記第２の電子証明書と前記第１の計算機から得られた前記セションデータを含めた第３の電子証明書を前記第１の計算機へ送信し、前記第１の計算機によって、前記第１の計算機によって生成された前記セションデータの少なくとも一部と、前記第２の電子証明書及び前記第３の電子証明書を署名検証により認証することを特徴とする認証処理方法。

【請求項４】

前記第２の計算機から暗号化された前記アプリケーションデータを受信した前記第３の計算機によって前記アプリケーションデータを復号化してデータチェックを行い、前記第１の計算機から暗号化された前記アプリケーションデータを受信した前記第３の計算機によって前記アプリケーションデータを復号化してデータチェックを行うことを特徴とする請求項３記載の認証処理方法。

【請求項５】

第１の計算機、前記第１の計算機に対してセション確立要求を行う第２の計算機および前記第１の計算機と前記第２の計算機との間に介在し両計算機間で送受信されるアプリケーションデータを中継する第３の計算機とを有するシステムであり、共通鍵暗号方式に基づいて前記第１の計算機と前記第２の計算機の間で前記アプリケーションデータを暗号化して送受信する暗号化通信システムにおいて、

前記第２の計算機からの当該セション固有のセションデータを含むセション確立要求に対し、生成した当該セション固有のセションデータを付加して前記第１の計算機へセション確立要求を送信する前記第３の計算機に設けられる手段、生成した当該セション固有のセションデータと前記第２の計算機および前記第３の計算機から得られた前記セションデータとを含めた第１の電子証明書を前記第３の計算機へ送信する前記第１の計算機に設けられる手段、受信した前記第１の電子証明書を署名検証により認証後に前記第１の電子証明書と前記第１の計算機および前記第２の計算機から得られた前記セションデータを含めた第２の電子証明書を前記第２の計算機へ送信する前記第３の計算機に設けられる手段、および前記第１の計算機、前記第２の計算機および前記第３の計算機によって生成された当該セション固有のセションデータの少なくとも一部と、前記第１の電子証明書及び前記第２の電子証明書を署名検証により認証する前記第２の計算機に設けられる手段を有する

ことを特徴とする認証処理システム。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
本発明は、計算機ネットワークを使用し暗号化通信を行うシステムにおける認証処理技術に関する。
【０００２】
【従来の技術】
インターネット、イントラネットに代表される計算機ネットワークにおいて、クレジットカード番号や個人情報などの機密データをクライアントからサーバに送信する場合には、データの盗聴を防止するためにデータを共通鍵暗号方式により暗号化処理することが一般的に行われている。またデータの改ざんを検出するためにメッセージに対してほとんど衝突することのないハッシュ関数で処理したハッシュ値を付加する処理が行なわれている。これら暗号処理やハッシュ関数処理に使用される鍵などのデータを交換するのに用いられる公開鍵を取得するために、暗号化通信を行う前にＸ．５０９証明書などの電子証明書を利用した認証処理も行われている。
【０００３】
データ暗号化処理、ハッシュ関数処理、認証処理を行う場合には、特開平１１－２７２６１５号公報に記載のようにＳＳＬ（Ｓｅｃｕｒｅ Ｓｏｃｋｅｔｓ Ｌａｙｅｒ）や、その後継であるＴＬＳ（Ｔｒａｎｓｐｏｒｔ Ｌａｙｅｒ Ｓｅｃｕｒｉｔｙ）に代表される通信手順が広く使用されている。これらの通信手順では、クライアントとサーバとの間で直接的に認証処理、鍵交換処理を実行してセションを確立し、暗号化通信を行う。そのため両者の間にプロキシまたはゲートウエイが介在していた場合には、特開２０００－３１２２０３号公報に記載のように、プロキシまたはゲートウエイが暗号化されたデータをそのまま転送することによりクライアントとサーバが直接通信する方法や、特開２０００－１７４７９７号公報に記載のように、クライアントとプロキシまたはゲートウエイとの間、プロキシまたはゲートウエイとサーバとの間で別々のセションを確立し暗号化通信を行う方法が知られている。
【０００４】
【発明が解決しようとする課題】
上記従来技術において、プロキシまたはゲートウエイが暗号化されたデータをそのまま転送する通信方法の場合には、クライアントは直接サーバを認証する、またはサーバは直接クライアントを認証することが可能であるため、通信相手が他者のなりすましではないことを確認することができるものの、プロキシまたはゲートウエイでは暗号化データを復号することが出来ないため、通信内容をチェックし意図されない情報の流入、流出を防止することはできない。
【０００５】
またクライアントとプロキシまたはゲートウエイとの間、プロキシまたはゲートウエイとサーバとの間で別々のセションを確立する通信方法の場合には、暗号化データはプロキシまたはゲートウエイにおいて一度復号されるため通信内容に対するチェックが可能であるものの、クライアント、サーバはプロキシまたはゲートウエイを認証するのみであり、クライアントがサーバを直接認証する、またはサーバがクライアントを直接認証することはできない。
【０００６】
本発明は、クライアントとサーバがプロキシまたはゲートウエイを介して別々のセションにより通信を行っている場合において、クライアントとサーバとの直接認証を可能とし、かつプロキシまたはゲートウエイにおいて暗号化データを復号することによる通信内容のチェックを可能とすることを目的とする。
【０００７】
【課題を解決するための手段】

上記目的を達成するために、まずクライアントは、暗号化通信セションの確立要求をプロキシまたはゲートウエイに送信する。プロキシまたはゲートウエイは、そのセションに対する乱数などのセション固有データを生成してクライアントのセション確立要求に追加し、さらにサーバに送信する。サーバは暗号化通信セション確立要求を受信すると、一時的公開鍵ペア（公開鍵、私有鍵）を生成し、そのうちの公開鍵と、受信したセション確立要求に含まれているセションデータと、サーバが新規に作成したセションデータと、サーバ自身の電子証明書を含めた一時的電子証明書を発行する。そしてこの一時的電子証明書をプロキシまたはゲートウエイに送信する。プロキシまたはゲートウエイでは、受信した一時的電子証明書の検証後、サーバの電子証明書を取り出し検証処理を行う。検証に成功した場合には、プロキシまたはゲートウエイは、クライアントとの間のセションデータと受信した一時的電子証明書を含めた一時的電子証明書をサーバと同様の手順により発行する。この一時的電子証明書とプロキシまたはゲートウエイ自身の電子証明書をクライアントへ送信する。

【０００８】

クライアントは、プロキシまたはゲートウエイと同様の検証処理をサーバの電子証明書が認証されるまで繰り返し、クライアントが生成したセションデータがサーバとプロキシまたはゲートウエイの双方で署名されていることを確認することで、クライアントはサーバとプロキシまたはゲートウエイの認証を行うことができる。サーバがクライアントの認証を要求している場合には、クライアントはサーバが生成したセションデータを含んだ同様の一時的電子証明書を作成しプロキシまたはゲートウエイに送信する。プロキシまたはゲートウエイは一時的電子証明書検証処理後、サーバとの間のセションデータを含めた一時的電子証明書を生成しサーバに送信する。サーバは上記と同様の署名検証処理を行うことで、サーバはプロキシまたはゲートウエイが介在していた場合においてもクライアントを認証することができる。

【０００９】

【発明の実施の形態】

本発明をＳＳＬを使用した場合の例について、図面を用いて詳細に説明する。ＳＳＬプロトコルは、暗号化通信処理を行うレコード層と、その上位に位置するハンドシェイクプロトコル、Ｃｈａｎｇｅ Ｃｉｐｈｅｒ Ｓｐｅｃプロトコル、アラートプロトコル、および送受信されるアプリケーションデータを有する上位層とから構成される通信手順である。レコード層の上位に位置するそれぞれのプロトコルは、クライアントとサーバとの間で交換される単数または複数のメッセージから構成されている。ＳＳＬでは、レコード層の上位に位置するプロトコルまたはデータ型を新たに追加することが可能となっているが、本発明においては主となる４つのプロトコルまたはデータ型を使用した場合について説明する。

【００１０】

ハンドシェイクプロトコルは、交換される電子証明書と暗号化もしくは署名されたデータとを使用して通信相手を認証し、さらにその電子証明書に記載されている公開鍵暗号方式の公開鍵または一時的に作成された公開鍵ペアの内の公開鍵を使用して、プレマスターシークレットを交換し、このデータから生成した共通鍵暗号方式で使用する秘密鍵などのデータをクライアントとサーバとの間で共有することで、セションの確立を行うものである。通信相手の認証方法には、サーバのみ認証を行う場合、サーバとクライアントの相互の認証を行う場合、そして双方とも認証を行わない場合がある。本発明においては、サーバまたはクライアントの認証を行なう場合を前提とする。

【００１１】

ハンドシェイクプロトコルメッセージ交換中において、メッセージ交換処理に使用されている共通鍵暗号やハッシュ関数の種別の変更を通知するために使用されるのがＣｈａｎｇｅ Ｃｉｐｈｅｒ Ｓｐｅｃプロトコルである。メッセージ交換処理中にエラーが発生したときは、アラートプロトコルを使用してエラー情報の交換が行われる。ハンドシェイクプロトコルにおいて共有された共通鍵暗号方式の秘密鍵を使用した暗号化通信はレコード層

において実現される。ＳＳＬセションが確立された後、クライアントまたはサーバから送信しようとするアプリケーションデータはレコード層により暗号化されて送信され、サーバまたはクライアントが受信したデータはレコード層により復号される。以下の説明では、クライアントとサーバの間にあるプロキシまたはゲートウエイが１つの場合を示すが、複数あった場合においても同様の方法が適用される。

【００１２】

図１は、本発明の実施例におけるシーケンスを表す図である。クライアントとサーバがＳＳＬを使用した通信を行う場合には、クライアントは最初にＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２を送信することによって、セション確立要求を行う。このメッセージには、図２に示すようにＳＳＬのバージョン番号２０２、クライアントが生成した乱数２０４、以前のセションの再利用を行う場合にはそのセションＩＤ２０６、クライアントが使用を望んでいる共通鍵暗号方式とハッシュ関数のリスト２０８、および圧縮アルゴリズムのリスト２１０が含まれる。セションの再利用ではなく、新規セションを確立する場合には、セションＩＤ２０６を送信しない。

【００１３】

プロキシまたはゲートウエイに代表される計算機内の代理応答プログラムは、このＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２を受信すると、付加データを作成し、これを付加データ２１２としてＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２の最後に付加し、サーバに対し同様にＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０４を送信する。プロキシまたはゲートウエイによる付加データ２１２は、図３に示されるように、付加データであることを示す識別子３０２、シーケンス番号３０６の先頭から乱数３０８の最後尾までのデータ長を表すデータ長３０４、クライアントが送信したＣｌｉｅｎｔ Ｈｅｌｌｏメッセージに対してデータを付加した順番を表すシーケンス番号３０６、及び乱数３０８から構成される。乱数３０８は、プロキシまたはゲートウエイが生成した乱数である。この付加データ２１２は、サーバまたはクライアントに対しそのプロキシまたはゲートウエイがＣｌｉｅｎｔ Ｈｅｌｌｏメッセージを受信し中継したことを示すものである。

【００１４】

シーケンス番号３０６により、クライアントからサーバまでの間に、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージを中継したプロキシまたはゲートウエイの個数が明示される。ＳＳＬバージョン番号２０２は、クライアントがサポートしているバージョン以下で、かつプロキシまたはゲートウエイが使用可能な最大の値を格納し送信する。クライアントから受信したクライアント生成乱数２０４とセションＩＤ２０６は、そのまま送信する。共通鍵暗号、ハッシュ関数リスト２０８と圧縮アルゴリズムリスト２１０については、プロキシまたはゲートウエイは、そのプロキシまたはゲートウエイが使用出来ない暗号方式を除いたリストを再構成し、送信する。リストが空になった場合は通信不可とされる。

【００１５】

サーバがこのＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０４を受信すると、セションＩＤの有無によってセションの再利用を要求しているかどうか、要求している場合には再利用が可能であるかどうかをチェックする。再利用を行わない場合には、新規にセションＩＤを生成し、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１０６を生成してプロキシまたはゲートウエイへ送信する。セションＩＤは乱数から生成するのが一般的であるが、時刻など他のデータを元にしたものでもよい。本発明では特にどちらでも構わない。サーバは受信した共通鍵暗号、ハッシュ関数リスト２０８と圧縮アルゴリズムリスト２１０から、１つの共通鍵暗号、ハッシュ関数と圧縮アルゴリズムを選択する。

【００１６】

図４は、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１０６、１１６のデータ構成を示す図である。ＳＳＬバージョン４０２はサーバが選択したバージョン、サーバ作成乱数４０４はサーバが新規に作成した乱数、セションＩＤ４０６はサーバが作成したセションＩＤ、共通鍵暗号とハッシュ関数４０８及び圧縮アルゴリズム４１０はそれぞれサーバが選択した方式である。

【００１７】
クライアントがプロキシまたはゲートウエイを介在してサーバの認証を行う場合には、サーバは一時的電子証明書を作成し、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１０８としてプロキシまたはゲートウエイに送信する。
【００１８】
図５は、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１０８、１１８及びＣｌｉｅｎｔ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１２６，１３６として送信される一時的電子証明書５０２のデータ構成を示す図である。一時的電子証明書５０２は、発行元が管理する一時的電子証明書のシリアル番号５０４、電子証明書発行者名５０６、有効期限５０８、電子証明書の所有者名５１０、公開鍵５１２、セションデータ５１４、固定の電子証明書または一時的電子証明書５１６及び署名アルゴリズム名と署名５１８から構成される。公開鍵５１２は一時的に作成された公開鍵ペアのうちの公開鍵である。署名アルゴリズム名と署名５１８は、電子証明書の発行者が保持する電子証明書に記載されている公開鍵に対応する私有鍵で署名した署名データを含む。サーバが発行する一時的電子証明書の所有者名５１０は、固定の電子証明書の所有者名とは別名であるが、サーバが一時的に作成した証明書であることが明白となる名称を使用する。
【００１９】
Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１０８中のセションデータ５１４は、受信したクライアント作成乱数２０４、プロキシまたはゲートウエイが生成したシーケンス番号３０６と生成した乱数３０８、サーバ作成乱数４０４、選択した共通鍵暗号、ハッシュ関数４０８及び圧縮アルゴリズム４１０を含む。Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１１２を送信する場合には、セションデータ５１４にその発行者のリストを含める。この一時的電子証明書に対して署名５１８によってサーバが署名を行うことで、クライアント、プロキシまたはゲートウエイでは、送信したＣｌｉｅｎｔ Ｈｅｌｌｏメッセージのデータがサーバまで不正な改ざんやなりすましの存在なく伝達され、また伝達経路にはシーケンス番号３０６で示された数のプロキシまたはゲートウエイが存在していることを検証できる。さらにクライアントが要求している暗号以外の暗号で通信を行おうとしていないことや、サーバがクライアントに対して要求している認証内容も確認できる。
【００２０】
さらに送信した一時的電子証明書に記載されたデータでは共通鍵暗号方式の鍵を共有するのに不十分な場合には、Ｓｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１１０を送信する。このメッセージで送信される一時的公開鍵ペアは、一時的電子証明書に使用されたものとは別の鍵ペアを作成して使用する。サーバがクライアントの認証を要求している場合には、Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１１２を送信する。最後に、Ｓｅｒｖｅｒ Ｈｅｌｌｏ Ｄｏｎｅメッセージ１１４を送信し、サーバ側のメッセージ終了を通知する。
【００２１】
プロキシまたはゲートウエイは、サーバから受信した一連のメッセージに対する認証処理を行う。認証処理が完了し受理可能であるときには、サーバと同様に一時的な公開鍵ペアを作成し、一時的電子証明書５０２を発行する。一時的電子証明書５０２は、作成した公開鍵５１２、サーバから受信した一時的電子証明書５１６、プロキシまたはゲートウエイの電子証明書に記載の公開鍵に対応する私有鍵で署名した署名データ５１８を含む。一時的電子証明書５０２にはこの他に、プロキシまたはゲートウエイが管理する一時的電子証明書のシリアル番号５０４、プロキシまたはゲートウエイが発行する一時的電子証明書の発行者名５０６、有効期限５０８、一時的電子証明書の所有者名５１０が含まれる。ここでの一時的電子証明書の所有者名５１０は、サーバ証明書の所有者名を使用する。またセションデータ５１４は、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２で受信したクライアント作成乱数２０４、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１０６で受信したサーバ作成乱数４０４、プロキシまたはゲートウエイが選択した共通鍵暗号、ハッシュ関数４０８と

圧縮アルゴリズム４１０を含む。Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１１２を受信したときには、セションデータ５１４に、受信した発行者のリストからプロキシまたはゲートウエイが受理不可能な発行者のリストを除いた残りのリストを含める。

【００２２】

そしてクライアントに対しＳｅｒｖｅｒ Ｈｅｌｌｏメッセージ１１６、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１１８を送信する。サーバからＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１１０を受信しているときには、一時的公開鍵ペアをさらに生成し、その内の公開鍵を使用したＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１２０を送信する。サーバからＣｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１１２を受信しているときには、前述のように発行者リストが変更されたＣｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１２２を送信する。最後に、Ｓｅｒｖｅｒ Ｈｅｌｌｏ Ｄｏｎｅメッセージ１２４を送信する。Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１１８では、生成した一時的電子証明書とともに、プロキシまたはゲートウエイの電子証明書も同時に送信する。サーバから受信したＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージに含まれている公開鍵は、プロキシまたはゲートウエイで保持し、クライアントには送信しない。

【００２３】

クライアントは、プロキシまたはゲートウエイから受信した一連のメッセージに対する認証を行う。認証処理が完了し受理可能である場合には、一連の応答メッセージを送信する。プロキシまたはゲートウエイからＣｅｒｔｉｆｉｃａｔｅＲｅｑｕｅｓｔメッセージ１２２を受信しており、かつ送信可能な証明書を保持している場合には、クライアントはサーバの場合と同様な方法により一時的電子証明書５０２を発行する。

【００２４】

クライアントが発行する一時的電子証明書５０２は、クライアントが管理するシリアル番号５０４、クライアントの電子証明書記載の公開鍵に対応する私有鍵で処理した署名データ５１８を記載する。またセションデータ５１４にＳｅｒｖｅｒ Ｈｅｌｌｏメッセージ１１６で受信したサーバ作成乱数４０４、セションＩＤ４０６、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１１８で受信した一時的電子証明書５０２に格納されているすべての電子証明書に署名したプロキシまたはゲートウエイまたはサーバの名称リスト、Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１２２で受信したＣＡ（認証局）のリストを含める。一時的電子証明書の所有者名５１０は、固定のクライアント電子証明書の所有者名とは別名であるが、クライアントが一時的に作成した証明書であることが明白となる名称を使用する。

【００２５】

一時的電子証明書を発行後、これをＣｌｉｅｎｔ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１２６として送信する。そしてＣｌｉｅｎｔ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１２８、Ｃｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージ１３０、Ｃｈａｎｇｅ Ｃｉｐｈｅｒ Ｓｐｅｃメッセージ１３２、Ｆｉｎｉｓｈｅｄメッセージ１３４を送信する。

【００２６】

Ｃｌｉｅｎｔ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１２８では、ＳｅｒｖｅｒＣｅｒｔｉｆｉｃａｔｅメッセージ１１８で受信した一時的電子証明書５０２に記載の公開鍵またはＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１２０により受信した公開鍵を使用する。プロキシまたはゲートウエイは、一時的電子証明書５０２作成時における一時的鍵ペアのうちの私有鍵またはＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージで送信した公開鍵に対応する私有鍵を使用する。またＣｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージ１３０の送信時には、クライアントは一時的電子証明書５０２を作成したときの一時的鍵ペアのうちの私有鍵を使用してデータに署名処理を行う。プロキシまたはゲートウエイは、Ｃｌｉｅｎｔ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１２６で受信した一時的電子証明書に記載の公開鍵を使用して、Ｃｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージデータの検証を行う。

【００２７】
プロキシまたはゲートウエイは、クライアントからの一連のメッセージの認証処理を行う。認証処理が完了し受理可能である場合には、クライアントの場合と同様に一時的電子証明書５０２を発行する。この場合一時的な公開鍵ペアは、新規に作成しても、前述のＳｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１１８を送信する際に作成したものと同じものを使用してもよいが、シリアル番号５０４は異なる値を使用する。セションデータ５１４として記載するデータは、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１０６で受信したサーバ作成乱数４０４、セションＩＤ４０６、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１０８で受信した一時的電子証明書５０２に格納されているすべての電子証明書に署名したプロキシまたはゲートウエイまたはサーバの名称リスト、Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ１１２で受信したＣＡ（Ｃｅｒｔｉｆｉｃａｔｉｏｎ Ａｕｔｈｏｒｉｔｙ）のリストである。一時的証明書の所有者名５１０は、クライアント電子証明書の所有者名を使用する。

【００２８】
プロキシまたはゲートウエイは、クライアントが発行した一時的電子証明書５０２を含むプロキシまたはゲートウエイの一時的電子証明書をＣｌｉｅｎｔ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１３６として送信する。そしてＣｌｉｅｎｔ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１３８、Ｃｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージ１４０、Ｃｈａｎｇｅ Ｃｉｐｈｅｒ Ｓｐｅｃメッセージ１４２、Ｆｉｎｉｓｈｅｄメッセージ１４４を送信する。Ｃｌｉｅｎｔ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１３８では、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１０８で受信した一時的電子証明書５０２に記載の公開鍵またはＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１１０により受信した公開鍵を使用する。サーバは、一時的電子証明書５０２作成時における一時的鍵ペアのうちの私有鍵またはＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージで送信した公開鍵に対応する私有鍵を使用する。またＣｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージ１４０の送信時には、プロキシまたはゲートウエイにおいて一時的電子証明書５０２を作成したときの一時的鍵ペアのうちの私有鍵を使用してデータに署名処理を行う。サーバは、Ｃｌｉｅｎｔ Ｃｅｒｔｉｆｉｃａｔｅメッセージ１３６で受信した一時的電子証明書５０２に記載の公開鍵を使用して、Ｃｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージデータ１４０の検証を行う。

【００２９】
サーバは、プロキシまたはゲートウエイからの一連のメッセージの認証処理を行う。認証処理が完了し受理可能である場合には、その応答としてＣｈａｎｇｅＣｉｐｈｅｒ Ｓｐｅｃメッセージ１４６、Ｆｉｎｉｓｈｅｄメッセージ１４８を送信する。Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０４を受信した際、セションＩＤの再利用を行うことに決定した場合においては、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１０６送信後すぐにこれらのメッセージの送信処理を行う。

【００３０】
プロキシまたはゲートウエイは、これらのメッセージを受信し検証が完了すると、同様にクライアントへＣｈａｎｇｅ Ｃｉｐｈｅｒ Ｓｐｅｃメッセージ１５０及びＦｉｎｉｓｈｅｄメッセージ１５２を送信する。サーバがセション再利用を行った場合には、プロキシまたはゲートウエイもセション再利用を行い、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１１６を送信後すぐにこれらのメッセージを送信する。プロキシまたはゲートウエイでセションの再利用が出来ない場合には、クライアントに対してＨｅｌｌｏ Ｒｅｑｕｅｓｔメッセージを送信し、新規セションの生成を行う。クライアントにおいて、Ｆｉｎｉｓｈｅｄメッセージ１５２の検証が完了すると、ＳＳＬセションが確立したことになる。この後、アプリケーションデータの送受信を行う。

【００３１】
なおＨｅｌｌｏ Ｒｅｑｕｅｓｔメッセージまたはアラートプロトコルメッセージをサーバが送信した場合には、プロキシまたはゲートウエイはメッセージ受信後すぐにクライア

ントにこのＨｅｌｌｏ Ｒｅｑｕｅｓｔメッセージまたはアラートプロトコルメッセージを送信する。プロキシまたはゲートウエイからＨｅｌｌｏＲｅｑｕｅｓｔメッセージの送信を開始することもできる。プロキシまたはゲートウエイからアラートプロトコルメッセージを送信するときは、クライアントとサーバの両方に対して送信を行う。

【００３２】

図６は、セション確立後の処理シーケンスを示す図である。クライアントは、プロキシまたはゲートウエイと共有している鍵を使用して送信データを暗号化処理し、その暗号化データ６０２をプロキシまたはゲートウエイへ送信する。プロキシまたはゲートウエイは、クライアントと共有している鍵を使用して受信データを復号し、データの内容をチェックした後、サーバと共有している鍵を使用して暗号化処理し、その暗号化データ６０４をサーバへ送信する。サーバは、プロキシまたはゲートウエイと共有している鍵を使用して受信データを復号し、受信データの処理を行う。その処理結果は、プロキシまたはゲートウエイと共有している鍵を使用して暗号化処理し、その暗号化データ６０６をプロキシまたはゲートウエイへ送信する。プロキシまたはゲートウエイは、サーバと共有している鍵を使用して受信データを復号し、データの内容をチェックした後、クライアントと共有している鍵を使用して暗号化処理し、その暗号化データ６０８をクライアントへ送信する。クライアントではプロキシまたはゲートウエイと共有している鍵を使用して受信データを復号し処理結果を得る。

【００３３】

図７は、実施形態のシステムの構成図である。計算機７０２は、記憶装置７１６、この記憶装置７１６に格納されるクライアントプログラム７０４およびデータ格納装置７１８を有する。クライアントプログラム７０４は、ＳＳＬ通信処理部７０６、電子証明書認証処理部７０８、電子証明書生成処理部７１０、乱数発生処理部７１２、データ処理部７１４を有する。クライアントプログラム７０４又はその一部を計算機７０２が読み取り可能な記録媒体に格納し、その駆動装置を介して記憶装置７１６に読み込み、計算機７０２によって実行することが可能である。

【００３４】

ＳＳＬ通信処理部７０６は、記憶装置７１６から取り出した以前のセションＩＤ２０６とそのセションデータや、乱数生成処理部７１２により生成した乱数２０４を使用したＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２などのＳＳＬメッセージ送信処理、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ１１６などのＳＳＬメッセージ受信処理、一時的電子証明書５０２に含まれている公開鍵５１２やＳｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ１２０により取得した公開鍵を使用した暗号化処理、ハッシュ関数を使用したハッシュ処理、一時的電子証明書５０２の公開鍵５１２に対応する私有鍵での署名処理、共有された鍵を使用したデータ暗号化、復号処理を行う。

【００３５】

電子証明書検証処理部７０８は、データ格納装置７１８から取得したＣＡの電子証明書とＳＳＬ通信処理部７０６で取得したＳＳＬメッセージを使用して、受信した一時的電子証明書５０２やサーバ電子証明書５１６の検証処理を行う。電子証明書生成処理部７１０は、サーバがクライアントの認証を要求している場合において、乱数生成処理部７１２から取得した乱数などを使用した一時的公開鍵ペアの生成と、ＳＳＬ通信処理部７０６で取得したＳＳＬメッセージから一時的電子証明書５０２の作成を行う。一時的電子証明書生成時には、クライアントの電子証明書に記載の公開鍵に対応する私有鍵での署名処理を行う。一時的電子証明書５０２は、暗号化処理セションが無効化されるまで、記憶装置７１６またはデータ格納装置７１８に格納される。データ処理部７１４は、処理を行うためのアプリケーションデータをＳＳＬ通信処理部７０６に渡し、またサーバから取得された結果データの処理を行う。

【００３６】

クライアントプログラム７０４は、ネットワーク７２０を通じて計算機７２２内で動作しているプロキシまたはゲートウエイプログラム７２４との通信を行う。計算機７２２には

、記憶装置７３６、記憶装置７３６上に格納されるプロキシまたはゲートウエイプログラム７２４およびデータ格納装置７３８を有する。プロキシまたはゲートウエイプログラム７２４は、ＳＳＬ通信処理部７２６、電子証明書認証処理部７２８、電子証明書生成処理部７３０、乱数生成処理部７３２を有する。これらはそれぞれ、ＳＳＬ通信処理部７０６、電子証明書認証処理部７０８、電子証明書生成処理部７１０および乱数生成処理部７１２と同様の処理を行う。ただし電子証明書生成処理部７３０は、プロキシまたはゲートウエイの電子証明書に記載の公開鍵に対応する私有鍵で署名処理を行う。プロキシまたはゲートウエイプログラム７２４にはさらにデータ処理部７３４があり、ここではＳＳＬセションが確立しアプリケーションデータをクライアントまたはサーバから受信した際、そのデータチェック処理を行う。プロキシまたはゲートウエイプログラム７２４又はその一部を計算機７２２が読み取り可能な記録媒体に格納し、その駆動装置を介して記憶装置７３６に読み込み、計算機７２２によって実行することが可能である。

【００３７】

プロキシまたはゲートウエイプログラム７２４は、ネットワーク７４０を通じて計算機７４２内で動作しているサーバプログラム７４４との通信を行う。計算機７４２は、記憶装置７５６、記憶装置７５６に格納されるサーバプログラム７４４およびデータ格納装置７５８を有する。サーバプログラム７４４は、ＳＳＬ通信処理部７４６、電子証明書検証処理部７４８、電子証明書生成処理部７５０、乱数生成処理部７５２を有する。これらはそれぞれ、ＳＳＬ通信処理部７０６、電子証明書検証処理部７０８、電子証明書生成処理部７１０および乱数生成処理部７１２と同様の処理を行う。ただし電子証明書生成処理部７５２は、サーバの電子証明書に記載の公開鍵に対応する私有鍵で署名処理を行う。サーバプログラム７４４にはさらにデータ処理部７５４があり、ここではＳＳＬセションが確立しアプリケーションデータをクライアント、プロキシまたはゲートウエイから受信した際、そのデータ処理を行う。サーバプログラム７４４又はその一部を計算機７４２が読み取り可能な記録媒体に格納し、その駆動装置を介して記憶装置７５６に読み込み、計算機７４２によって実行することが可能である。

【００３８】

図８及び図９は、クライアントプログラム７０４のフローチャートである。クライアントプログラム７０４を起動しＳＳＬによるアクセスを行うと、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２で使用する乱数２０４を生成し、セション再利用する場合には記憶装置７１６からセションＩＤ２０６を取り出す（ステップ８０２）。セションの再利用を行わない場合にはセションＩＤの取り出し処理を行わない。次にＣｌｉｅｎｔ Ｈｅｌｌｏメッセージ１０２を送信し、プロキシまたはゲートウエイからの応答待ち（ステップ８０４）となる。このとき応答にはアラートメッセージ、セション再利用メッセージ群、サーバメッセージ群受信の３つがある。セション再利用メッセージ群とは、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ、Ｃｈａｎｇｅ Ｃｉｐｈｅｒ ＳｐｅｃメッセージとＦｉｎｉｓｈｅｄメッセージを意味する。サーバメッセージ群とは、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージ、Ｓｅｒｖｅｒ Ｃｅｒｔｉｆｉｃａｔｅメッセージ、Ｓｅｒｖｅｒ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ、Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｑｕｅｓｔメッセージ、Ｓｅｒｖｅｒ Ｈｅｌｌｏ Ｄｏｎｅメッセージもしくはその一部を意味する。

【００３９】

ここでアラートメッセージを受信した場合（ステップ８０６）には、処理を終了する。セション再利用メッセージ群を受信した場合（ステップ８０８）では、そのセションに関連するデータを記憶装置７１６から取り出す処理（ステップ８２２）を行い、受信したＦｉｎｉｓｈｅｄメッセージの検証を行った後、Ｆｉｎｉｓｈｅｄメッセージ群の送信処理（ステップ８２４）を行う。Ｆｉｎｉｓｈｅｄメッセージ群とは、Ｃｈａｎｇｅ Ｃｉｐｈｅｒ ＳｐｅｃメッセージとＦｉｎｉｓｈｅｄメッセージを意味する。その後、データを暗号化してプロキシまたはゲートウエイ、そしてサーバに送信し、受信データを復号し結果を得る処理（ステップ８１８）を行う。

【００４０】

ＳＳＬメッセージ受信待ち（ステップ８０４）においてサーバメッセージ群の受信（ステップ８１０）があった場合には、図９で示される認証処理手順を行った後、クライアントメッセージ群を送信し、Ｆｉｎｉｓｈｅｄメッセージ群の受信待ち（ステップ８１２）となる。クライアントメッセージ群とは、Ｃｌｉｅｎｔ Ｃｅｒｔｉｆｉｃａｔｅメッセージ、Ｃｌｉｅｎｔ Ｋｅｙ Ｅｘｃｈａｎｇｅメッセージ、Ｃｅｒｔｉｆｉｃａｔｅ Ｖｅｒｉｆｙメッセージ、Ｃｈａｎｇｅ Ｃｉｐｈｅｒ Ｓｐｅｃメッセージ、Ｆｉｎｉｓｈｅｄメッセージまたはその一部を意味する。プロキシまたはゲートウエイから有効なＦｉｎｉｓｈｅｄメッセージ群を受信すると、セション確立（ステップ８１４）に成功したこととなり、そのセションデータを記憶装置７１６またはデータ格納装置７１８に格納する処理（ステップ８１６）を行う。その後、データの暗号化、送受信、復号により結果を得る処理（ステップ８１８）を行う。

【００４１】

セション確立（ステップ８１４）において、無効なＦｉｎｉｓｈｅｄメッセージを受信したなど、セション確立に失敗したときには、アラートメッセージ送信処理（ステップ８２０）を行って処理を終了する。さらにＳＳＬメッセージ受信待ち（ステップ８０４）において、アラートメッセージ、セション再利用メッセージ、サーバメッセージ群、ＨｅｌｌｏＲｅｑｕｅｓｔメッセージ以外のメッセージを受信した場合においても、アラートメッセージ送信処理（ステップ８２０）を行い処理を終了する。

【００４２】

ＳＳＬメッセージ受信待ち処理（ステップ８０４）において、サーバメッセージ群の受信があった場合（ステップ８１０ＹＥＳ）には、図９に移り、電子証明書の認証処理（ステップ９０２）を行う。認証処理の詳細は図１４において記述する。電子証明書の認証に失敗した場合（ステップ９０４ＮＯ）には、アラートメッセージを送信（ステップ９１２）し、ＳＳＬ処理を終了する。電子証明書の認証に成功した場合（ステップ９０４ＹＥＳ）には、クライアントはサーバの認証に成功し本発明の目的の１つを達成したことになる。そしてサーバからクライアントの認証要求がある場合（ステップ９０６あり）には、一時的電子証明書を受信しているか否かの判断（ステップ９０８）を行う。受信している場合には、クライアントとサーバとの間にプロキシまたはゲートウエイが介在しているため、同様に一時的電子証明書の生成処理（ステップ９１０）を行う。一時的電子証明書ではなく通常の電子証明書である場合には、クライアントとサーバとの間にはプロキシが介在していないため、従来の処理を行う。またクライアントの認証要求がない場合（ステップ９０６なし）にも、一時的電子証明書生成処理（ステップ９１０）は不要である。

【００４３】

図１０および図１１は、プロキシまたはゲートウエイプログラム７２４のフローチャートである。プロキシまたはゲートウエイプログラム７２４は起動されるとＳＳＬメッセージの受信待ち（ステップ１００２）となる。このときに受信するメッセージは、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージ、サーバメッセージ群、クライアントメッセージ群、セション再利用メッセージ群、Ｆｉｎｉｓｈｅｄメッセージ群、アラートメッセージ、データ送受信である。

【００４４】

Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージを受信したとき（ステップ１００４）には、乱数３０８を生成し付加データ２１２を作成してクライアントから受信したＣｌｉｅｎｔ Ｈｅｌｌｏメッセージに付加する処理（ステップ１００６）を行い、その送信処理（ステップ１００８）を行う。このときＳＳＬバージョン番号２０２はクライアントが使用可能なバージョン以下でかつプロキシまたはゲートウエイが使用可能な最大の値を格納し、共通鍵暗号、ハッシュ関数リスト２０８と圧縮アルゴリズムリスト２１０にはプロキシまたはゲートウエイが適用することの出来ないものを削除した残りを格納して送信する。クライアント作成乱数２０４、セションＩＤ２０６はそのまま送信する。

【００４５】

ＳＳＬメッセージ受信待ち処理（ステップ１００２）においてサーバメッセージ群または

クライアントメッセージ群を受信した場合（ステップ１０１０ＹＥＳ）には、図１１で示される電子証明書認証処理、一時的電子証明書生成処理を行った後、同じメッセージ群を送信する。

【００４６】

ＳＳＬメッセージ受信待ち処理（ステップ１００２）においてセション再利用メッセージ群を受信した場合（ステップ１０１２ＹＥＳ）には、プロキシまたはゲートウエイに格納してるセションデータの取り出し処理（ステップ１０１４）を行い、これを使用してセション再利用メッセージ群送信処理（ステップ１０１６）を行う。セション再利用に失敗した場合には、クライアントとプロキシまたはゲートウエイ、プロキシまたはゲートウエイとサーバ間のどちらの側においても新規セションの確立を行う。

【００４７】

ＳＳＬメッセージ受信待ち処理（ステップ１００２）においてＦｉｎｉｓｈｅｄメッセージ群を受信した場合（ステップ１０１８ＹＥＳ）には、メッセージの検証を行った後、サーバと同様にクライアントに対するＦｉｎｉｓｈｅｄメッセージ群送信処理（ステップ１０２０）を行い、確立されたセションデータの格納処理（ステップ１０２２）を行う。またアラートメッセージを受信した場合（ステップ１０２４ＹＥＳ）には、そのままアラートメッセージの送信処理（ステップ１０２６）を行う。

【００４８】

そしてセション確立後に共通鍵暗号方式による暗号データ受信があった場合（ステップ１０２８ＹＥＳ）には、データを復号し、データチェック処理などを行った後、通信先の鍵を使用してデータを再び暗号化し送信（ステップ１０３０）する。これにより本発明における目的の１つが実現されることになる。

【００４９】

上記のＳＳＬメッセージ以外のデータを受信した場合には、アラートメッセージ送信処理（ステップ１０３２）を行いＳＳＬメッセージ受信待ち状態（ステップ１００２）に戻る。

【００５０】

ＳＳＬメッセージ受信待ち（ステップ１００２）において、サーバメッセージ群またはクライアントメッセージ群を受信した場合（ステップ１０１０ＹＥＳ）には、クライアントプログラムの場合と同様、まず図１１に示す電子証明書認証処理（ステップ１１０２）が行われる。認証処理の詳細は図１４において記述する。電子証明書の認証に失敗した場合（ステップ１１０４ＮＯ）には、アラートメッセージをサーバとクライアント双方への送信処理（ステップ１１１０）を行い、再びＳＳＬメッセージ受信待ち状態（ステップ１００２）に戻る。電子証明書の認証に成功した場合（ステップ１１０４ＹＥＳ）には、一時的電子証明書生成処理（ステップ１１０６）を行い、サーバメッセージ群またはクライアントメッセージ群を送信（ステップ１１０８）し、再びＳＳＬメッセージの待ち状態（ステップ１００２）に戻る。

【００５１】

図１２及び図１３はサーバプログラム７４４のフローチャートである。サーバプログラムは、起動するとＳＳＬリクエスト受信待ち状態（ステップ１２０２）となる。このときに受信するメッセージは、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージ、クライアントメッセージ群、データ送受信である。セション再利用を行った場合に受信するＦｉｎｉｓｈｅｄメッセージ群とアラートメッセージを受信したときには、メッセージ検証やコネクション切断などの対応処理を行うものの、ＳＳＬメッセージ受信待ち状態（ステップ１２０２）のままである。

【００５２】

ＳＳＬメッセージ受信待ち状態（ステップ１２０２）においてＣｌｉｅｎｔ Ｈｅｌｌｏメッセージの受信があった場合（ステップ１２０４ＹＥＳ）には、図１３に示される処理を行い、再びＳＳＬメッセージ受信待ち状態（ステップ１２０２）となる。クライアントメッセージ群の受信があった場合（ステップ１２１０ＹＥＳ）には、図１４において記述

される電子証明書の認証処理（ステップ１２１２）を行う。認証に失敗した場合（ステップ１２１４ＮＯ）には、アラートメッセージを送信（ステップ１２２０）してＳＳＬメッセージ受信待ち（ステップ１２０２）に戻る。認証に成功すると、Ｆｉｎｉｓｈｅｄメッセージ群送信処理（ステップ１２１６）を行い、セションデータ格納処理（ステップ１２１８）を行うことで、セション再利用を可能とする。

【００５３】

ＳＳＬメッセージ受信待ち状態（ステップ１２０２）において、セションが確立している場合に暗号化データ受信があった場合（ステップ１２０６ＹＥＳ）には、データを復号しデータ処理を行った後、処理結果を暗号化処理して送信（ステップ１２０８）し、再びＳＳＬメッセージ受信待ち状態（ステップ１２０２）に戻る。上記メッセージ以外のメッセージを受信した場合には、アラートメッセージを送信（ステップ１２２０）し再びＳＳＬメッセージ受信待ち状態（ステップ１２０２）に戻る。

【００５４】

ＳＳＬメッセージ受信待ち状態（ステップ１２０２）において、ＣｌｉｅｎｔＨｅｌｌｏメッセージを受信したとき（ステップ１２０４ＹＥＳ）には、図１３に移り、まずセション再利用要求がされているか否かのチェック処理（ステップ１３０２）を行う。再利用要求がある場合には、それが可能か否かのチェック処理（ステップ１３１２）を行う。再利用が可能である場合には、記憶装置７５６からセションデータ取り出し処理（ステップ１３１４）を行い、これを使用してセション再利用メッセージ群送信処理（ステップ１３１６）を行う。再利用が不可能である場合には、再利用要求がない場合と同じく、新規にセションを生成する。セションを生成する場合、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージにおいて送信するサーバ作成乱数４０４とセションＩＤ４０６の生成処理（ステップ１３０４）を行う。そしてＣｌｉｅｎｔ Ｈｅｌｌｏメッセージを調べ、プロキシまたはゲートウエイによる付加データ２１２が存在しているか否かを判別（ステップ１３０６）する。付加されていれば、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージはプロキシを経由したものであるため、一時的電子証明書生成処理（ステップ１３０８）を行う。この処理は図１５において記述する。付加データ２１２のない場合には、一時的電子証明書生成処理は行わずに、サーバメッセージ群送信処理（ステップ１３１０）を行う。

【００５５】

図１４は、電子証明書の認証処理のフローチャートである。電子証明書は、ここではクライアントまたはサーバ電子証明書、そして一時的電子証明書の両方のことである。

【００５６】

電子証明書認証ではまず、電子証明書発行者名５０６に記載されている発行者の電子証明書の取得処理（ステップ１４０２）を行う。取得は、ＳＳＬメッセージで送信されたものを用いる場合と、データ格納装置７１８、７３８、７５８に格納されているものを取り出す場合がある。取得できた場合、さらにその電子証明書の発行者の電子証明書を取得する。これをルートＣＡの電子証明書にたどり着くまで繰り返す。ただしルートＣＡの電子証明書は、ＳＳＬメッセージで送信されたものは使用せず、データ格納装置内から取得したもののみを使用する。電子証明書の取得が１つでも失敗した場合（ステップ１４０６）は、認証処理は失敗したものとし処理を終了する。すべての電子証明書が取得可能である場合（ステップ１４０４ＹＥＳ）には、電子証明書記載の公開鍵を使用して証明書連鎖の検証を行う。この連鎖の検証に失敗した場合（ステップ１４１０）は、認証処理は失敗したものとし処理を終了する。すべての検証に成功した場合（ステップ１４０８ＹＥＳ）には、証明書記載のデータの検証を行う。これには証明書が有効期限内での使用であるか、証明書の失効情報であるＣＲＬ（Ｃｅｒｔｉｆｉｃａｔｅ Ｒｅｖｏｃａｔｉｏｎ Ｌｉｓｔ）にそのシリアル番号が記載されていないかなどの処理が含まれる。このデータ検証に失敗した場合（ステップ１４１４）には、認証処理は失敗したものとし処理を終了する。データ検証に成功した場合（ステップ１４１２ＹＥＳ）では、電子証明書の種別を判別（ステップ１４１６）し、これがサーバまたはクライアントの電子証明書であったならば認証は成功（ステップ１４１８）したものとなる。

【００５７】
一時的電子証明書の場合には、その内部にセションデータと電子証明書が含まれているため、さらにこれらの検証処理を行う。セションデータ検証処理では、記載されているデータがＳＳＬメッセージで受信したものと同一であるか、共通鍵暗号の場合には要求しているものと異なった暗号種別の選択が行われていないか、プロキシまたはゲートウエイによる付加データ２１２のシーケンス番号は順番通りか、順番が抜けていたり追加されていたりしていないかなどの検証を行う。検証に失敗した場合（ステップ１４２２）には、認証処理は失敗したものとして処理を終了する。セションデータ検証処理に成功した場合（ステップ１４２０ＹＥＳ）には、内部に電子証明書が含まれているか否かをチェックする処理（ステップ１４２４）を行う。含まれていない場合には、認証処理は失敗（ステップ１４２６）したものとして処理を終了する。含まれていた場合には、それを取り出し（ステップ１４２８）、再び電子証明書の発行者からルートＣＡまでの電子証明書獲得処理（ステップ１４０２）から認証処理を行っていく。

【００５８】
図１５は、一時的電子証明書の生成手順を示すフローチャートである。電子証明書を生成するには、必要なデータを取得し署名計算をしたものを、電子証明書フォーマットとして編集する。処理においては、まず発行する証明書のシリアル番号の取得処理（ステップ１５０２）、発行者自身の名称取得処理（ステップ１５０４）を行う。そして電子証明書の有効期限の計算処理（ステップ１５０６）を行う。さらに発行対象者の名称取得処理（ステップ１５０８）を行う。プロキシまたはゲートウエイでは、この名称はサーバまたはクライアントの電子証明書に記載の名称を使用する。サーバまたはクライアントでは、自身の名称ではなく、自身を示す別の名称を使用する。これらを行った後、一時的公開鍵ペア計算処理（ステップ１５１０）、セションデータ取得処理（ステップ１５１２）、格納する電子証明書取得処理（ステップ１５１４）と続き、これらをフォーマットしたデータに対して発行者の電子証明書に記載の公開鍵に対応する私有鍵で署名処理（ステップ１５１６）を行い、上記データをフォーマットに合わせて編集し終了する。

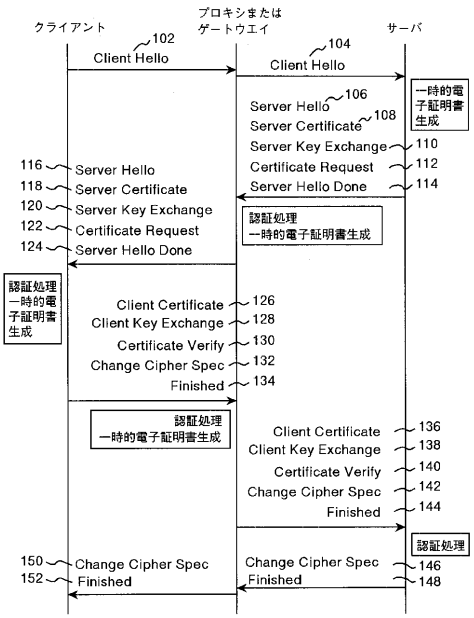
【００５９】
【発明の効果】
以上説明したように本発明によれば、一時的な電子証明書の内部にそのセションでしか取得できない情報とクライアントまたはサーバ証明書を格納し署名することによって、情報の取得が証明されると同時にプロキシまたはゲートウエイの連鎖がクライアントまたはサーバから明白となる。本発明を実施するに当り、ＳＳＬを使用した通信方法の定義をほとんど変更することがない。また本発明によって、認証されていない攻撃者がその間に入り込むことや、ＳＳＬメッセージの改ざん、暗号化データの盗聴、改ざんが防止またはクライアントまたはサーバから検出可能な状態のままで、クライアント、プロキシまたはゲートウエイ、そしてサーバの間で電子証明書に基づく認証が正しく行われ、その結果プロキシまたはゲートウエイにおいて通信内容のチェックが可能となる。

【図面の簡単な説明】
【図１】本発明の実施例を表すシーケンス図の一例である。
【図２】ＳＳＬにおけるＣｌｉｅｎｔ Ｈｅｌｌｏメッセージで送信されるデータ構成図である。
【図３】Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージに対してプロキシまたはゲートウエイが付加するデータの構成図である。
【図４】ＳＳＬにおけるＳｅｒｖｅｒ Ｈｅｌｌｏメッセージで送信されるデータ構成図である。
【図５】本発明で使用する一時的電子証明書の構成図の一例である。
【図６】ＳＳＬセション確立後に行われるデータ通信を表すシーケンス図である。
【図７】本発明のシステム構成図の一例である。
【図８】クライアントプログラムの処理手順を示すフローチャートである。
【図９】クライアントプログラムにおいて、Ｓｅｒｖｅｒ ＨｅｌｌｏメッセージからＳ

ｅｒｖｅｒ Ｈｅｌｌｏ Ｄｏｎｅメッセージまでの一連のメッセージを受信したときのフローチャートである。
【図１０】プロキシまたはゲートウエイプログラムの処理手順を示すフローチャートである。
【図１１】プロキシまたはゲートウエイプログラムにおいて、Ｓｅｒｖｅｒ Ｈｅｌｌｏメッセージからｓｅｒｖｅｒ Ｈｅｌｌｏ Ｄｏｎｅまでの一連のメッセージ、Ｃｌｉｅｎｔ ＣｅｒｔｉｆｉｃａｔｅメッセージからＦｉｎｉｓｈｅｄメッセージまでの一連のメッセージを受信したときのフローチャートである。
【図１２】サーバプログラムの処理手順を示すフローチャートである。
【図１３】サーバプログラムにおいて、Ｃｌｉｅｎｔ Ｈｅｌｌｏメッセージを受信したときのフローチャートである。
【図１４】電子証明書の認証処理手順例を示すフローチャートである。
【図１５】一時的電子証明書の生成手順例を示すフローチャートである。
【符号の説明】
２１２：プロキシまたはゲートウエイ付加データ、５１４：セションデータ、５１６：一時的電子証明書に格納された電子証明書、７１０，７３０，７５０：電子証明書生成処理部

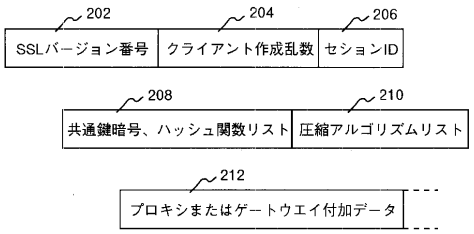
【図１】

図 1



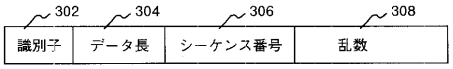
【図２】

図 2



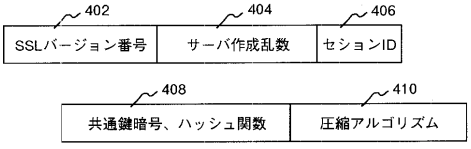
【図３】

図 3



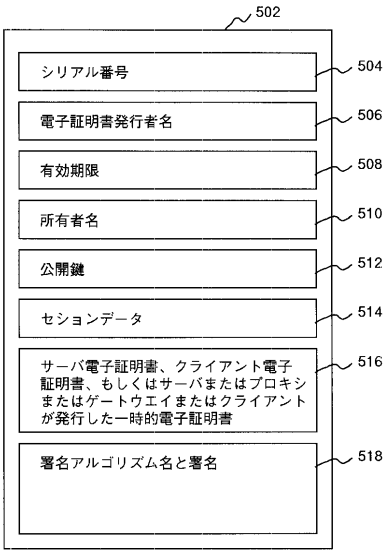
【図４】

図 4



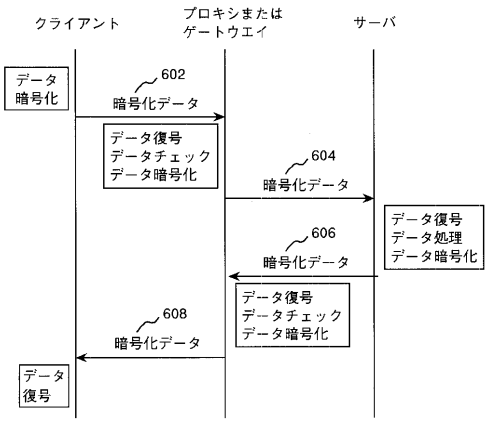
【図５】

図 5



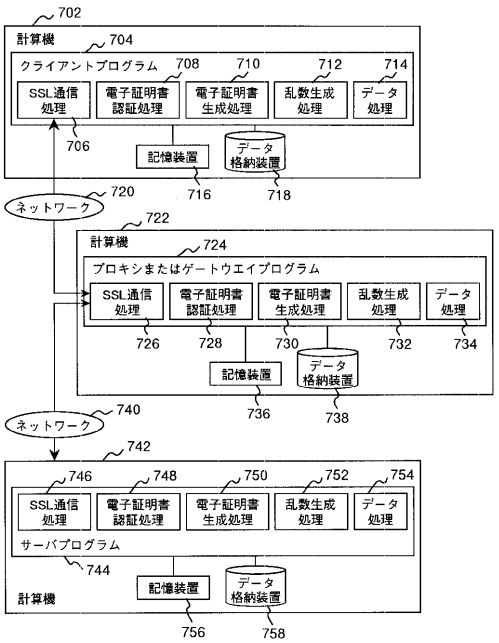
【図６】

図 6



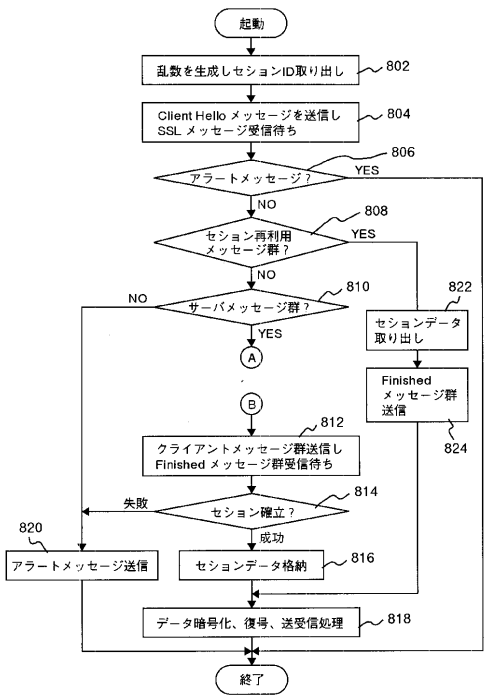
【図７】

図 7



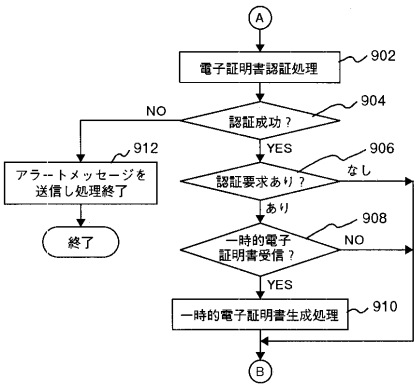
【図８】

図 8



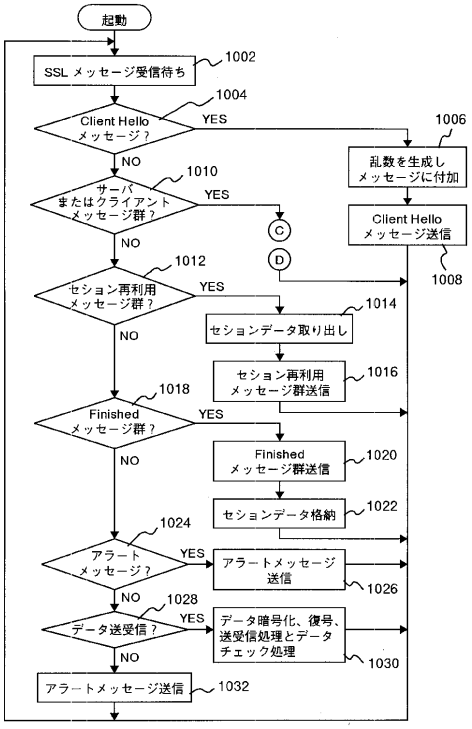
【図９】

図 9



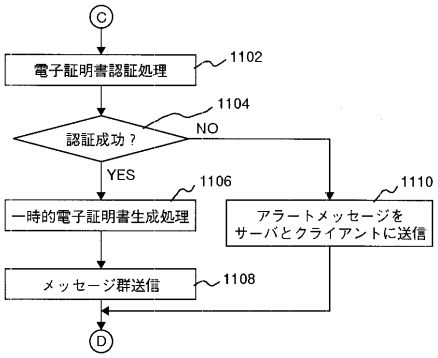
【図１０】

図 １０



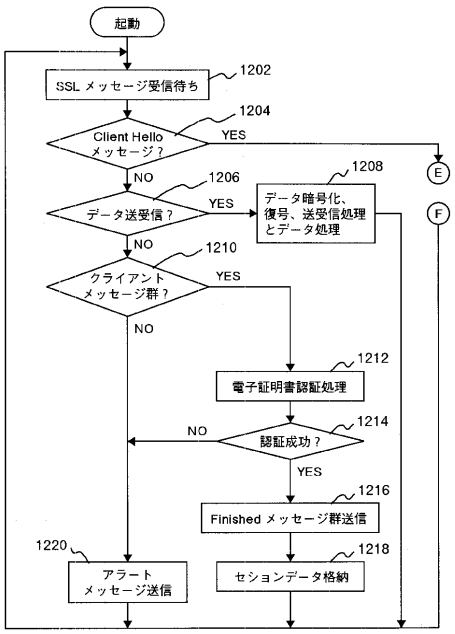
【図１１】

図 １１



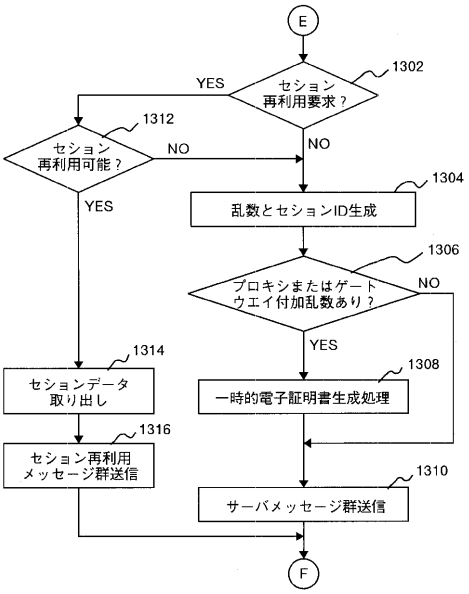
【図１２】

図 １２



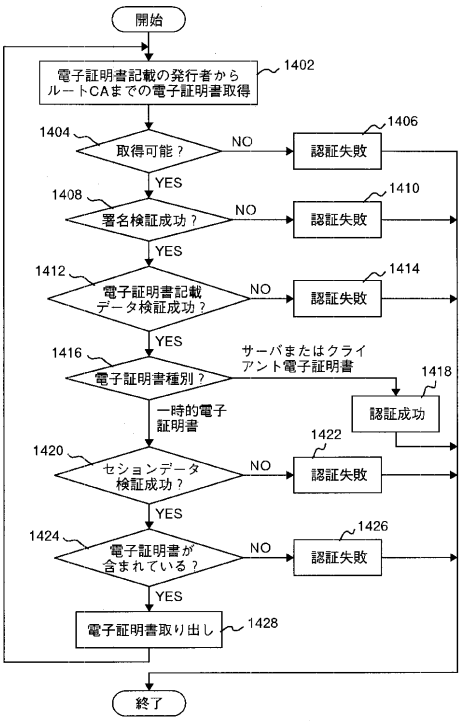
【図１３】

図 １３



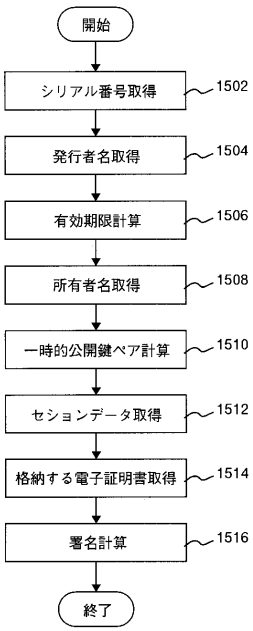
【図１４】

図 １４



【図１５】

図 １５



フロントページの続き

審査官　石川　正二

(56)参考文献　国際公開第０１／００２９３５（WO，Ａ１）
光来健一，千葉滋，インターネットにおけるパーソナルネットワークの構築，情報処理学会研究報告 「システムソフトウェアとオペレーティング・システム」，２００１年　7月２７日，Vol．2001，No．78，pp．83-90

(58)調査した分野(Int.Cl.，ＤＢ名)
H04L 9/32
G06F 21/20