

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4145118号
(P4145118)

(45) 発行日 平成20年9月3日 (2008.9.3)

(24) 登録日 平成20年6月27日 (2008.6.27)

(51) Int.Cl.		F I			
G 0 6 F	21/22	(2006.01)	G 0 6 F	9/06	6 6 O G
G 0 6 K	17/00	(2006.01)	G 0 6 K	17/00	L
G 0 6 K	19/00	(2006.01)	G 0 6 K	19/00	Q
H 0 4 L	9/10	(2006.01)	H 0 4 L	9/00	6 2 1 Z

請求項の数 23 (全 56 頁)

(21) 出願番号 特願2002-321844 (P2002-321844)
(22) 出願日 平成14年11月5日 (2002.11.5)
(65) 公開番号 特開2003-223235 (P2003-223235A)
(43) 公開日 平成15年8月8日 (2003.8.8)
審査請求日 平成17年10月28日 (2005.10.28)
(31) 優先権主張番号 特願2001-359940 (P2001-359940)
(32) 優先日 平成13年11月26日 (2001.11.26)
(33) 優先権主張国 日本国 (JP)

(73) 特許権者 000005821
松下電器産業株式会社
大阪府門真市大字門真1006番地
(74) 代理人 100109553
弁理士 工藤 一郎
(72) 発明者 峰村 淳
大阪府門真市大字門真1006番地 松下
電器産業株式会社内

審査官 宮司 卓佳

最終頁に続く

(54) 【発明の名称】 アプリケーション認証システム

(57) 【特許請求の範囲】

【請求項 1】

端末と、認証モジュールと、からなるアプリケーション認証システムであって、
端末は、
アプリケーションをダウンロードするダウンロード部と、
認証モジュールに自身が認証されることを条件としてアプリケーションの認証のための
処理をする T R M アクセスライブラリ部と、
を有し、

認証モジュールは、
T R M アクセスライブラリ部を認証するための情報である T R M アクセスライブラリ部
認証情報を耐タンパ領域に保持する T R M 部と、
T R M アクセスライブラリ部認証情報に基づいて端末の T R M アクセスライブラリ部を
認証する T R M アクセスライブラリ部認証部と、
を有するアプリケーション認証システム。

【請求項 2】

耐タンパ領域に情報を保持しその情報を用いて認証のための処理を行なう認証モジュール
を備えた端末であって、
アプリケーションをダウンロードするダウンロード部と、
認証モジュールに自身が認証されることを条件としてアプリケーションの認証のための
処理をする T R M アクセスライブラリ部と、を有し、

10

20

前記認証モジュールは、

前記 T R M アクセスライブラリ部を認証するための情報である T R M アクセスライブラリ部認証情報を前記耐タンパ領域に保持する T R M 部と、

T R M アクセスライブラリ部認証情報に基づいて前記 T R M アクセスライブラリ部を認証する T R M アクセスライブラリ部認証部と、を有する端末。

【請求項 3】

前記ダウンロード部は、改ざんのないことを認証するために用いる情報である署名が付加されたアプリケーションをダウンロードし、

前記 T R M アクセスライブラリ部は、ダウンロード部にダウンロードされたアプリケーションから署名認証用ダイジェストを生成し、生成した署名認証用ダイジェストと、署名と、を含む署名認証情報を認証モジュールに出力する署名認証情報出力部をさらに有し、

前記認証モジュールは、署名認証情報出力部から出力された署名認証情報を入力する署名認証情報入力部と、

前記署名認証情報入力部から入力される署名認証用ダイジェストと、署名と、に基づいて署名の検証を行う署名認証部と、をさらに有する請求項 2 記載の端末。

【請求項 4】

前記認証モジュールは、

前記署名認証情報入力部から入力される署名を利用して署名由来ダイジェストを生成するための署名由来ダイジェスト生成情報を取得する署名由来ダイジェスト生成情報取得部と、

前記署名認証情報入力部から入力された署名と、前記署名由来ダイジェスト生成情報取得部に保持された署名由来ダイジェスト生成情報と、を利用して署名由来ダイジェストを生成する署名由来ダイジェスト生成部とをさらに有し、

前記署名認証部は、前記署名由来ダイジェスト生成部で生成された署名由来ダイジェストと、前記署名認証情報入力部から入力された署名認証用ダイジェストと、に基づいて認証を行う請求項 3 に記載の端末。

【請求項 5】

前記認証モジュールを認証するための認証モジュール認証部を有する請求項 2 から 4 のいずれかに記載の端末。

【請求項 6】

前記 T R M アクセスライブラリ部は、認証されたアプリケーションに対して利用を認めるリソースに関する情報であるアプリケーション利用リソース情報を保持するアプリケーション利用リソース情報保持手段を有する請求項 5 に記載の端末。

【請求項 7】

前記 T R M アクセスライブラリ部は、前記認証モジュール認証部による認証がされた認証モジュールの T R M 部に対してアプリケーション利用リソース情報を出力するアプリケーション利用リソース情報出力手段をさらに有し、

前記認証モジュールの T R M 部は、該端末の T R M アクセスライブラリ部のアプリケーション利用リソース情報出力手段から出力されたアプリケーション利用リソース情報を耐タンパ領域に書き換え可能に保持する請求項 5 または 6 に記載の端末。

【請求項 8】

前記 T R M アクセスライブラリ部は、アプリケーション利用リソース情報に基づいて、アプリケーションに対して、リソースの利用を認める請求項 6 または 7 記載の端末。

【請求項 9】

署名が付されたアプリケーション利用リソース情報をダウンロードするアプリケーション利用リソース情報ダウンロード部をさらに有し、

前記 T R M アクセスライブラリ部は、アプリケーション利用リソース情報ダウンロード部にダウンロードされたアプリケーション利用リソース情報に付された署名を検証することを特徴とする請求項 6 から 8 に記載の端末。

【請求項 10】

10

20

30

40

50

署名が付されたアプリケーション利用リソース情報をダウンロードするアプリケーション利用リソース情報ダウンロード部をさらに有し、

前記 T R M アクセスライブラリ部は、アプリケーション利用リソース情報ダウンロード部にダウンロードされたアプリケーション利用リソース情報から署名認証用ダイジェストを生成し、

生成した署名認証用ダイジェストと、署名と、を含む署名認証情報を認証モジュールに出力するアプリケーション利用リソース情報署名認証情報出力部をさらに有し、

前記認証モジュールは、前記アプリケーション利用リソース情報署名認証情報出力部から出力された署名認証情報を入力するアプリケーション利用リソース情報署名認証情報入力部と、

前記アプリケーション利用リソース情報署名認証情報入力部から入力される署名認証用ダイジェストと、署名と、に基づいて署名の検証を行なうアプリケーション利用リソース情報署名認証部と、をさらに有する請求項 6 から 8 に記載の端末。

【請求項 1 1】

前記 T R M アクセスライブラリ部認証情報が、該端末に固有の情報である請求項 2 記載の端末。

【請求項 1 2】

前記認証モジュールの T R M 部にアクセスをする端末アプリケーションを保持する端末アプリケーション保持部をさらに有し、

前記認証モジュールの T R M 部は、耐タンパ領域に、認証モジュール内にて動作する認証モジュール内アプリケーションを保持する認証モジュール内アプリケーション保持部をさらに有し、

前記認証モジュール内アプリケーションは、T R M アクセスライブラリ部認証部による T R M アクセスライブラリ部の認証の成功を条件として端末アプリケーションからのアクセスを受け入れて動作する請求項 2 から 4 のいずれかに記載の端末。

【請求項 1 3】

前記認証モジュールの T R M 部は、前記 T R M アクセスライブラリ部認証部による前記 T R M アクセスライブラリ部の認証の成功を条件として認証結果識別子を生成する認証結果識別子生成手段を有し、

認証モジュール内アプリケーションは、認証を示す認証結果識別子の存在を条件として端末アプリケーションに対して認証モジュール内アプリケーションに対するアクセスを可能とし、認証モジュール内アプリケーションは端末アプリケーションからのアクセスを受け入れる請求項 1 2 に記載の端末。

【請求項 1 4】

前記認証モジュールの T R M 部は、前記 T R M アクセスライブラリ部によるアプリケーションの認証の成功を条件としてアプリ認証結果識別子を生成するアプリ認証結果識別子生成手段を有し、

認証モジュール内アプリケーションは、認証の成功を示すアプリ認証結果識別子の存在を条件として端末アプリケーションに対して認証モジュール内アプリケーションに対するアクセスを可能とし、認証モジュール内アプリケーションは端末アプリケーションからのアクセスを受け入れる請求項 1 2 に記載の端末。

【請求項 1 5】

端末と、認証モジュールと、サーバと、からなるアプリケーション認証システムであって、

端末は、

アプリケーションをダウンロードするダウンロード部と、

認証モジュールに自身が認証されることを条件としてアプリケーションの認証のための処理をする T R M アクセスライブラリ部と、を有し

認証モジュールは、

T R M アクセスライブラリ部を認証するための情報である T R M アクセスライブラリ部

10

20

30

40

50

認証情報を耐タンパ領域に保持する T R M 部と、

T R M アクセスライブラリ部認証情報に基づいて端末の T R M アクセスライブラリ部を認証する T R M アクセスライブラリ部認証部と、を有し、

サーバは、

端末の T R M アクセスライブラリ部を介する認証モジュールの T R M 部の認証を行い、これが成功することを条件として T R M アクセスライブラリ部の認証が成功したと判断するサーバ T R M アクセスライブラリ部認証部を有するアプリケーション認証システム。

【請求項 16】

認証モジュールと、

署名が付されたアプリケーションが動作する端末であり前記認証モジュールによる認証が成功すると前記アプリケーションの認証に用いる署名用ダイジェストを生成する T R M アクセスライブラリ部を備える端末と共に用いられるサーバであって、

前記 T R M アクセスライブラリ部を介する前記 T R M 部の認証を行い、これが成功した場合に限り前記 T R M アクセスライブラリ部の認証が成功したと判断するサーバ T R M アクセスライブラリ部認証部と、

前記サーバ T R M アクセスライブラリ部認証部による前記認証モジュールの認証が成功すると、前記認証モジュールによって認証された前記端末の T R M アクセスライブラリ部から、前記 T R M アクセスライブラリ部が生成した前記署名用ダイジェストと当該アプリケーションに付された署名とを受信するアプリ認証データ入力部と、

前記アプリ認証データ入力部が受信した署名用ダイジェストと、署名と、に基づいて前記端末上にダウンロードしたアプリケーションの認証を行なうサーバアプリ認証部と、を有するサーバ。

【請求項 17】

認証モジュールと、

アプリケーションが動作する端末であり前記認証モジュールに認証されると前記アプリケーションの認証処理を行う T R M アクセスライブラリ部を備える端末と共に用いられるサーバであって、

端末の T R M アクセスライブラリ部を介する認証モジュールの T R M 部の認証を行い、これが成功することを条件として T R M アクセスライブラリ部の認証が成功したと判断するサーバ T R M アクセスライブラリ部認証部と、

前記サーバ T R M アクセスライブラリ部認証部による前記認証モジュールの認証が成功すると、前記認証モジュールによって認証された前記端末の T R M アクセスライブラリ部から、前記 T R M アクセスライブラリ部が生成した前記端末上で動作するアプリケーションの認証の成功を示す認証成功情報を受信する認証成功情報入力部と、

前記認証成功情報入力部が受信した認証成功情報に基づいて前記端末上にダウンロードしたアプリケーションの認証を行なうサーバアプリ認証部と、を有するサーバ。

【請求項 18】

前記ダウンロード部は、アプリケーションが使用を許可または許可されないリソースを記述したアプリケーション利用リソース情報をダウンロードし、前記アプリケーションが前記リソースを使用する際、前記 T R M アクセスライブラリ部に対してリソース使用の要求を出し、

前記 T R M アクセスライブラリ部は、内部に保持している前記アプリケーション利用リソース情報を参照し、要求されたリソースが使用可能かどうかを判断し、使用可能であれば、要求されたリソースをアプリケーションに使用させる、請求項 1 に記載のアプリケーション認証システム。

【請求項 19】

ダウンロード部は、ダウンロードされたアプリケーションの実行時、又は / 及び、アプリケーションの認証時に、サーバより使用許諾書をダウンロードする請求項 1 に記載のアプリケーション認証システム。

【請求項 20】

10

20

30

40

50

ダウンロード部は、ダウンロードされたアプリケーションの実行時、又はノ及び、アプリケーションの認証時に、ダウンロードされた前記アプリケーション利用リソース情報の有効性をサーバに問い合わせる請求項１８に記載のアプリケーション認証システム。

【請求項２１】

第一の機器と、認証モジュールと、からなるアプリケーション認証システムであって、
第一の機器は、
アプリケーションを格納するアプリケーション格納部と、
認証モジュールに自身が認証されることを条件としてアプリケーションの認証のための処理をするＴＲＭアクセスライブラリ部と、を有し、

認証モジュールは、

ＴＲＭアクセスライブラリ部を認証するための情報であるＴＲＭアクセスライブラリ部認証情報を耐タンパ領域に保持するＴＲＭ部と、

ＴＲＭアクセスライブラリ部認証情報に基づいて第一の機器のＴＲＭアクセスライブラリ部を認証するＴＲＭアクセスライブラリ部認証部と、を有するアプリケーション認証システム。

【請求項２２】

耐タンパ領域に情報を保持しその情報を用いて認証のための処理を行なう認証モジュールを備えた端末で用いる認証方法であって、

前記端末は、認証モジュールとＴＲＭアクセスライブラリ部とを備え

前記認証モジュールは、前記ＴＲＭアクセスライブラリ部を認証するための情報であるＴＲＭアクセスライブラリ部認証情報を前記耐タンパ領域に保持するＴＲＭ部を備え、

前記認証方法は、

アプリケーションをダウンロードするダウンロードステップと、

前記認証モジュールにより、前記ＴＲＭアクセスライブラリ部認証情報に基づいて、前記ＴＲＭアクセスライブラリ部を認証するステップと、

前記認証モジュールに前記ＴＲＭアクセスライブラリ部が認証されることを条件として、当該ＴＲＭアクセスライブラリ部により、アプリケーションの認証のための処理をするステップと、

を有することを特徴とする認証方法。

【請求項２３】

耐タンパ領域に情報を保持しその情報を用いて認証のための処理を行なう認証モジュールを備えた端末で用いる認証プログラムであって、

前記端末は、認証モジュールとＴＲＭアクセスライブラリ部とを備え

前記認証モジュールは、前記ＴＲＭアクセスライブラリ部を認証するための情報であるＴＲＭアクセスライブラリ部認証情報を前記耐タンパ領域に保持するＴＲＭ部を備え、

前記認証プログラムは、

アプリケーションをダウンロードするダウンロードステップと、

前記認証モジュールにより、前記ＴＲＭアクセスライブラリ部認証情報に基づいて、前記ＴＲＭアクセスライブラリ部を認証するステップと、

前記認証モジュールに前記ＴＲＭアクセスライブラリ部が認証されることを条件として、当該ＴＲＭアクセスライブラリ部により、アプリケーションの認証のための処理をするステップと、

を前記端末に実行させることを特徴とする認証プログラム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、携帯電話などの端末とＩＣカードなどの認証モジュールとを有するシステムにおいて、端末と端末で動作するアプリケーションとを認証モジュールによって認証する技術に関する。

【０００２】

【従来の技術】

従来、ＩＣカードを端末に装着してサーバと商取引操作を行なうシステムにおいては、端末に耐タンパ性を持った領域を確保できないので、サーバ内で動作するアプリケーションプログラム（以下、「アプリケーション」と略す）が直接ＩＣカードを認証するようになっていた。従って、端末は、サーバとＩＣカードの間の通信を中継するのみであった。

【０００３】

一方、近年において、携帯電話などにサーバよりアプリケーションがダウンロードでき、携帯端末において動作することが可能となった。

【０００４】**【発明が解決しようとする課題】**

しかしながら、携帯電話などにダウンロードされたアプリケーションが不正な動作を行なう可能性があるため、ダウンロードされたアプリケーションの動作は、厳格に制限がされている。

【０００５】

例えば、携帯電話にダウンロードされたアプリケーションは、携帯電話に装着されたＩＣカードにデータを書き込むことができなかつたり、各種インターフェースの使用が制限（禁止）されたりなど、ローカルリソースの使用制限が大きい。

【０００６】

また、携帯電話にダウンロードされたアプリケーションは、携帯電話やＩＣカードなどが保持している電話帳やアドレス帳などに記録されたメールアドレスやメールの受信箱に格納されたメールの内容など、個人に関する情報などの読み書きができないように制限を受けている。これは、当該アプリケーションが正規のものであり、携帯電話やＩＣカードなどの内部に保持されている情報にアクセスする権利があるかどうか、また、不正な動作をしないかどうか、という検証ができないからである。

【０００７】

このことは、将来的に希望されている携帯ツールの用途汎用化（多様化）やＥ－コマス（ＥＣ）への応用には極めて大きな阻害要因となる。

【０００８】

このような制限を無くするためには、ダウンロードされたアプリケーションを認証してアプリケーションの素性を確認する必要がある。例えば、アプリケーションに第三者が付した署名をアプリケーションとともにダウンロードし、その署名と署名が正しいと検証するのに必要な情報をＩＣカードに提示して認証を行なうことが考えられる。しかし、携帯電話がアプリケーションと署名とをダウンロードした後に、その携帯電話により署名が正しいと判断するのに必要な情報（例えば、ハッシュ関数により生成されるダイジェスト）が生成されるため、ダウンロードされたアプリケーションに付された署名とは別のダミーな署名とその署名により検証ができるような操作が施されたダイジェストが携帯電話によりＩＣカードに提示されてしまう可能性がある。このため、ＩＣカードに提示された署名とダイジェストとが、実際にダウンロードされたアプリケーションのものであるとＩＣカードが信頼することができず、ダウンロードされたアプリケーションをＩＣカードが認証を行なうことができないという問題がある。

【０００９】

また、携帯電話などの端末にダウンロードされたアプリケーション（以下、「端末アプリケーション」と略す）がＩＣカードにアクセスできるようにして、そのＩＣカードに保護されるように格納されている情報を読み書きできるためには、同様に、ＩＣカードへアクセスを行なう端末アプリケーションの認証のための処理を、そのＩＣカードが行い、アクセスを許可しても良いかどうかを判断することが必要である。

【００１０】

ＩＣカードなどのセキュアデバイスが、それにアクセスを行なう端末アプリケーションを認証するための処理としては、従来は、セキュアデバイスが、そこに保持されているのと同様の秘密な情報を端末アプリケーションが持っているかどうかを判断する、という処理

10

20

30

40

50

が行なわれていた。しかし、端末には、耐タンパ性を持った領域などの秘密な情報を安全に保持するための領域も機能もない。このため、このような秘密な情報が漏洩する可能性があり、従来の方法では、漏洩してしまった情報を端末アプリケーションが使用している可能性が排除できず、セキュアデバイスが端末アプリケーションを厳密には認証することができない、という課題がある。

【 0 0 1 1 】

【課題を解決するための手段】

この問題を解決するために本発明においては、ＩＣカードなどの認証モジュール内の耐タンパ領域に、携帯電話などの端末のＲＯＭに書き込まれたプログラム（以降において、「ライブラリ」と書く場合がある）を認証するための情報を持たせ、認証モジュールは、端末のＲＯＭやＴＲＭなどの容易に書き換えができない領域（以下、ＲＯＭと略する）に書き込まれたライブラリの認証を行なうようにした。

10

【 0 0 1 2 】

このように認証されたライブラリが、ダウンロードされたアプリケーションの署名と共に署名が正しいと判断するのに必要な情報を自ら生成してＩＣカードに提示すれば、ＩＣカードにとっては、認証したライブラリにより提示された署名と情報が、実際にダウンロードされたアプリケーションのものであると信頼することができるようになり、端末にダウンロードされたアプリケーションをＩＣカードが認証できるようになる。結果として、認証されたアプリケーションがＩＣカードにデータを書き込むことなどが可能となり、従来の端末を用いた商取引の操作よりも複雑な操作が実現できるようになる。

20

【 0 0 1 3 】

また、このように、端末にダウンロードされたアプリケーションがＩＣカードにより認証されるようになることにより、ダウンロードされたアプリケーションに、端末の外部インターフェースの利用を許可することも可能となる。

【 0 0 1 4 】

また、本発明の実施には、ダウンロードされるアプリケーションの署名をダウンロードすることが必要である。そのために、アプリケーションの署名をアプリケーションとは別にダウンロードするのみならず、既存のデータ仕様にも対応できるように、アプリケーションの定義ファイルに格納することを可能とする。このため、署名をアプリケーションのダウンロードとは別にダウンロードすることが不要となる。

30

【 0 0 1 5 】

【発明の実施の形態】

本発明は、端末と認証モジュールとを含むアプリケーション認証システムに関するが、「端末」とは、携帯電話に代表される携帯可能な電子機器であってもよい。また、パーソナルコンピュータや街角に設置させる公衆端末のように携帯が事実上出来ないようなものであってもよく、次に説明する認証モジュールが装着可能な電子機器であり、内部でアプリケーションを動作することが可能なものである。以下に説明するように端末は、その内部に種々なる部を有するが、端末にＲＯＭ、ＲＡＭ、ＣＰＵを備えることにより、このような部は、ソフトウェアにより実現可能となる。

【 0 0 1 6 】

40

「認証モジュール」とは、内部に記憶領域を持ったものであり、（ＳＤ）メモ리카ード、ＩＣカード、スマートカードがあり、装着された端末からデータを入力するとそれに対する返答を返すという動作を行なう。また、認証モジュールは、耐タンパ領域と言って、そこに格納された情報の不正な読み出しおよび不正な書き換えを防止する領域がある。以下で説明する認証モジュール内の部のいくつかは、このようなＩＣカード内で動作するカードアプリケーションによって実現することもできる。

【 0 0 1 7 】

通常、端末に認証モジュールが装着され、端末と認証モジュールの間で電気回路が形成され、情報の交換が行なわれる。ただし、端末の本体と認証モジュールが装着される部分とが分離し、通信によって情報の交換が行われる形態もある。

50

【 0 0 1 8 】

なお、端末で動作するアプリケーションは、算術演算、ローカルインターフェースの利用、サーバへのアクセス、耐タンパ領域へのアクセス、外部メモリへのアクセスのいずれか一以上の動作を行なう。「ローカルインターフェース」とは、端末のもつ外部インターフェースであり、例えば、I r D A (赤外線通信のためのインターフェース)、B l u e t o o t h、その他の無線通信、及び有線通信のためのインターフェースなどが挙げることができる。

【 0 0 1 9 】

なお、上記においては、端末と認証モジュールとが容易に分離可能であるとしたが、認証モジュールが端末の回路に対して圧着がされたり半田付けなどがされたりすることにより端末の内部に備えられ容易に分離できないようになっていてもよい。

10

【 0 0 2 0 】

また、上記において、「端末」と記したが、本発明は、携帯可能な機器に限られることなく、端末を使用する代わりにパーソナルコンピュータやワークステーションなどを用いて実施することが可能である。

【 0 0 2 1 】

また、本発明は、複数の機器が直列に接続された状態で、一方の端の機器から他方の端の機器が格納するアプリケーションを認証するための処理に応用することも可能である。

【 0 0 2 2 】

(実施の形態 1)

20

図 1 は、本発明の実施の形態 1 に関するアプリケーション認証システムの機能ブロック図である。本実施の形態におけるアプリケーション認証システムは、端末 1 0 0 と認証モジュール 1 0 1 から構成される。

【 0 0 2 3 】

端末 1 0 0 は、アプリケーションをダウンロードするダウンロード部 1 0 2 を有する。「アプリケーションをダウンロードする」とは、ダウンロード部 1 0 2 の外部からアプリケーションの実行のためのデータを読み込むことである。ここに「アプリケーションの実行のためのデータ」とは、アプリケーションが端末 1 0 0 により直接実行可能なバイナリデータであれば、そのバイナリデータであり、アプリケーションが端末 1 0 0 によって解釈実行される言語で記述されたものであれば、その記述である。ダウンロード部は、アプリケーションを実行のためのデータを読み込み、そのデータを保持することを行なう。

30

【 0 0 2 4 】

認証モジュール 1 0 1 は、T R M 部 1 0 3 を有している。T R M 部 1 0 3 は、アプリケーションの認証の処理のための情報を耐タンパ領域に保持する。なお、「T R M」は、「T a m p e r R e s i s t a n t M o d u l e」の略である。「アプリケーション」とは、端末 1 0 0 のダウンロード部 1 0 2 にダウンロードされたアプリケーションであり、「アプリケーションの認証」とは、アプリケーションが、信頼のおける者によって発行されたかどうか、不正な動作をしないことの保証を受けているものであるかどうか、あるいは、信頼のおける者から発行されてから後の改ざん、又は、不正な動作をしないことが保証されてから後の改ざんがされていないかどうか、などアプリケーションが不正な動作をしないものであることを確認することである。「アプリケーションの認証の処理」とは、このような確認するための処理である。

40

【 0 0 2 5 】

この認証の処理の方法としては、通常、S H A - 1 (S e c u r e H a s h S t a n d a r d - 1) や M D 5 (M e s s a g e D i g e s t 5) などの、ハッシュ関数(「ハッシュ関数」のかわりに「要約関数」と呼ばれる場合もある)を用いて、アプリケーションの実行のためのデータを入力データとして処理して得られる結果データを求め、それを暗号化したもの(いわゆる「署名」)が用いられる。ここに「ハッシュ関数」とは、入力データを処理して得られる結果データが一致するような二つの異なる入力データを見つけることが計算量的に困難な結果を返す関数である。従って「アプリケーションの認証の

50

処理のための情報」とは、この署名そのもの、あるいは、署名を復号してハッシュ値を得るのに必要な復号鍵である。「耐タンパ領域」とは、認証モジュールの記憶領域であって、その記憶領域のデータを不正に読み出すことや、その記憶領域のデータを不正に書き換えることが困難な記憶領域である。そのような記憶領域は、例えば、その記憶領域にアクセスするためには、正しい手順を行なわないとアクセスできないハードウェアを経由しなければならないようにしたり、記憶領域に記憶されているデータが暗号化されているようにしたりして実現される。

【 0 0 2 6 】

アプリケーションの認証は、端末 1 0 0 がダウンロード部 1 0 2 にダウンロードされたアプリケーションのハッシュ値を求め、そのハッシュ値とともに署名を認証モジュールに提示し、認証モジュール 1 0 1 は、T R M 部 1 0 3 に保持された情報により、そのハッシュ値からその署名を生成することができるかどうか、あるいは、そのハッシュ値が署名を復号して得られるハッシュ値と一致するかどうかを調べる。また、端末 1 0 0 がダウンロード部にダウンロードされたアプリケーションそのものと署名を認証モジュールに提示し、アプリケーションそのものと署名との関係を調べて認証を行なうようにしてもよい。

10

【 0 0 2 7 】

このような構成のアプリケーション認証システムにより、ダウンロード部 1 0 2 にダウンロードされたアプリケーションを、認証モジュール 1 0 1 の T R M 部 1 0 3 に保持された情報により、認証することができるので、不正なアプリケーションがダウンロードされて端末 1 0 0 内で実行、または、認証モジュール 1 0 1 内に格納されてしまうことを防止することができる。

20

【 0 0 2 8 】

図 2 は、本実施の形態のアプリケーション認証システムの一例を示す。この例において、アプリケーション 2 0 1 が格納された状態で認証モジュール 1 0 1 をサービス提供会社 2 0 0 が配布を行なう。認証モジュール 1 0 1 を入手した人が端末 1 0 0 に認証モジュール 1 0 1 を装着すると、認証モジュール 1 0 1 から端末 1 0 0 のダウンロード部 1 0 2 にアプリケーション 2 0 1 がダウンロードされる（矢印 2 0 3）。ダウンロードされたアプリケーション 2 0 2 が、認証モジュール 1 0 1 の T R M 部に格納された情報により認証がされると、アプリケーション 2 0 2 が端末 1 0 0 内で動作をして、サービス提供会社 2 0 0 からサービスの提供を受ける（矢印 2 0 4）。

30

【 0 0 2 9 】

図 3 は、本実施の形態のアプリケーション認証システムの別の例を示す。この例において、サービス提供会社 2 0 0 は、端末 1 0 0 のダウンロード部 1 0 2 へアプリケーション 3 0 2 をダウンロードする（矢印 3 0 3）。アプリケーション 3 0 2 に対して、認証モジュール 1 0 1 の T R M 部に保持された情報により、認証が行なわれ、不正なアプリケーションでないことが確認されると、認証モジュール 1 0 1 へアプリケーション 3 0 1 として格納される（矢印 3 0 4）。その後、アプリケーション 3 0 1 は、必要に応じて、端末 1 0 0 のダウンロード部 1 0 2 にダウンロードされて端末 1 0 0 内で実行される。

【 0 0 3 0 】

図 2 と図 3 とにおいては、端末 1 0 0 内で実行されるアプリケーションは認証モジュールからダウンロードされるが、端末が接続したサーバよりダウンロードされ、認証モジュールの T R M 部に保持された情報により、認証が行なわれ、不正なアプリケーションでないことが確認されると、そのアプリケーションが端末内で実行されるようになっている例も挙げることができる。

40

【 0 0 3 1 】

なお、「端末」という語を用いたが、これは、携帯電話に代表される携帯可能な端末などに限定されることを意味しない。例えば、家庭用電化製品であってもよいし、いわゆる、情報家電やネット家電と呼ばれるものであってもよい。そのような製品を例示列挙すれば、エアコンディショナ、加湿器、除湿器、空気清浄機、電子レンジ、オーブン、冷蔵庫、食器洗い機、湯沸し器、アイロン、ズボンプレスサー、電気掃除機、洗濯機、乾燥機、電

50

気毛布、電気敷布、照明機器、テレビ受像機、ラジオ受信機、テープレコーダなどのオーディオ機器、カメラ、ＩＣレコーダ、電話機、ファクシミリ送受信機、コピー機、プリンター、スキャナー、パーソナルコンピュータ、などを挙げることができる。

【００３２】

（実施の形態２）

図４は、本発明の実施の形態２に関するアプリケーション認証システムの機能ブロック図を示し、アプリケーション認証システムは、端末１００と認証モジュール１０１とからなる。端末１００は、ダウンロード部１０２とＴＲＭアクセスライブラリ部４０１とを有する。認証モジュール１０１は、ＴＲＭ部１０３とＴＲＭアクセスライブラリ部認証部４０２とを有する。

10

【００３３】

ダウンロード部１０２は、アプリケーションをダウンロードする。

【００３４】

ＴＲＭアクセスライブラリ部４０１は、認証モジュール１０１に自身が認証されることを条件としてアプリケーションの認証のための処理をする。すなわち、ＴＲＭアクセスライブラリ部４０１は、まず、認証モジュール１０１に自身を認証させ、正しく認証された場合に、アプリケーションの認証のための処理を行なう。

【００３５】

ＴＲＭアクセスライブラリ部４０１が認証モジュール１０１に自身を認証させる方法としては、端末１００に固有な情報を認証モジュール１０１に出力し、認証モジュール１０１は、出力された情報が耐タンパ領域に格納された情報と適合するかどうかで認証を行なう方法がある。「端末１００に固有な情報」としては、１ 端末が携帯電話である場合には、その電話番号が挙げられる。また、端末１００に固有な情報の別のものとしては、２ 端末１００の種類を特定する識別子や、端末１００に付けられた製造番号などの個々の端末ごとに異なるものが付される識別子であってもよい。また、３ 端末１００にインストールされているアプリケーションの組み合わせに関する情報を用いてＴＲＭアクセスライブラリ部４０１を認証する方法がある。「端末１００にインストールされているアプリケーション」とは、端末１００内に備えられたアプリケーションを意味し、端末の外部からダウンロードされたアプリケーションや、端末のＲＯＭに格納されたアプリケーションである。この場合、ＴＲＭアクセスライブラリ部４０１は、認証モジュール１０１に、端末１００内にどのアプリケーションがインストールされているかの情報（例えば、インストールされているアプリケーションの識別子など）を出力し、認証モジュール１０１は、出力された情報が、耐タンパ領域に格納された情報と適合するかどうかで認証を行なう。あるいは、不正な端末の情報が耐タンパ領域に格納されており、不正な端末の情報が適合しないかどうかで認証を行なう。

20

30

【００３６】

上で挙げた３として例えば、サービスＡとサービスＢとの両方のサービスの提供を受けるための会員になっている者に、新しいサービスＣの提供がされんとする。この場合、サービスＡの提供を受けるためのアプリケーションＡと、サービスＢの提供を受けるためのアプリケーションＢと、が端末１００にインストールされていることを条件としてＴＲＭアクセスライブラリ部４０１が認証モジュール１０１により認証されることにすれば、端末にインストールしているサービスＡとサービスＢとの会員に対して、サービスＣの提供を受けるためのアプリケーションＣを提供することが可能となる。

40

【００３７】

また、ＴＲＭアクセスライブラリ部４０１を識別するための情報を用いてＴＲＭアクセスライブラリ部４０１を認証する方法もある。「ＴＲＭアクセスライブラリ部４０１を識別するための情報」とは、例えば、ＴＲＭアクセスライブラリ部４０１を構成するソフトウェアを識別するための情報であり、そのソフトウェアの名前やバージョン番号やシリアル番号などである。ＴＲＭアクセスライブラリ部４０１は、ＴＲＭアクセスライブラリ部４０１を識別するための情報を認証モジュールに出力し、認証モジュール１０１は、出力さ

50

れた情報が、耐タンパ領域に格納された情報と適合するかどうかで認証を行なう。

【 0 0 3 8 】

「アプリケーションの認証のための処理」とは、ダウンロード部 1 0 2 にダウンロードされたアプリケーションの認証のための処理である。T R M アクセスライブラリ部 4 0 1 がアプリケーションの認証に関する処理の一部を行なってもよい。また、T R M アクセスライブラリ部 4 0 1 は認証モジュールによって認証されているので、アプリケーションの認証に関する処理の全てを行なってもよい。

【 0 0 3 9 】

また、T R M アクセスライブラリ部 4 0 1 の行なう処理は、アプリケーションの認証のための処理に限定される必要はない。例えば、図 3 6 に示すように、更に、T R M アクセスライブラリ部 4 0 1 は、その内部にアプリケーションマネージャとデバイスドライバとを備え、それらの処理を行なうようになっていてもよい。

10

【 0 0 4 0 】

「アプリケーションマネージャ」とは、ダウンロード部 1 0 2 にダウンロードされたアプリケーションの起動、終了、サスペンドなど、アプリケーションの動作の制御を行なう機能を提供する。なお、アプリケーションマネージャの別の名称としては、例えば、「アプリケーション制御プログラム」を挙げることができる。

【 0 0 4 1 】

「デバイスドライバ」とは、認証モジュールとの入出力を管理するプログラムである。例えば、認証モジュールごとに異なる入出力のための操作の仕様を吸収し、同一のインターフェースの操作によってアプリケーションが入出力を行なえるようにするためのプログラムである。なお、デバイスドライバの別の名称としては、「認証モジュールアクセスプログラム」を挙げることができる。

20

【 0 0 4 2 】

T R M 部 1 0 3 は、T R M アクセスライブラリ部認証情報を耐タンパ領域に保持する。「T R M アクセスライブラリ部認証情報」とは、T R M アクセスライブラリ部 4 0 1 を認証するための情報である。既に説明したように、この情報は、携帯電話番号などの端末 1 0 0 に固有の情報である場合や、端末 1 0 0 にインストールされているアプリケーションの組み合わせに関する情報である場合や、T R M アクセスライブラリ部 4 0 1 を識別するための情報である場合などがある。T R M アクセスライブラリ部認証情報は耐タンパ領域に保持されるため、携帯電話番号、端末 1 0 0 にインストールされているアプリケーションを識別するための情報、T R M アクセスライブラリ部 4 0 1 を識別するための情報が暗号化されて保持されている場合もある。

30

【 0 0 4 3 】

T R M アクセスライブラリ部認証部 4 0 2 は、T R M アクセスライブラリ部認証情報に基づいて端末のT R M アクセスライブラリ部 4 0 1 を認証する。すなわち、T R M アクセスライブラリ部 4 0 1 から認証モジュールに出力される認証のための情報と、T R M 部 1 0 3 に保持されるT R M アクセスライブラリ部認証情報と、により、T R M アクセスライブラリ部 4 0 1 を認証することを行なう。

【 0 0 4 4 】

図 5 は、アプリケーションを本実施の形態のアプリケーション認証システムにおいて、アプリケーションの認証の方法を説明するための図である。図 4 と図 5 とに示されたアプリケーション認証システムの違いは、図 5 の端末 1 0 0 のダウンロード部 1 0 2 が、改ざんのないことを認証するために用いる情報である署名が付加されたアプリケーションをダウンロードする点と、端末 1 0 0 が署名認証情報出力部 5 0 1 をさらに有する点と、認証モジュール 1 0 1 が署名認証情報入力部 5 0 2 と、署名認証部 5 0 3 とをさらに有する点である。

40

【 0 0 4 5 】

「改ざんのないことを認証するために用いる情報である署名」とは、アプリケーションが改ざんされていないことを確認するための情報である。図 6 は、アプリケーションと署名

50

の関係とを示す。アプリケーション本体 6 0 1 をデータとみなしてそれに対して、S H A - 1 や M D 5 などのハッシュ関数を適用して得られる値を暗号化したものが署名 6 0 2 である。署名 6 0 2 を用いてアプリケーション本体が改ざんされていないことを確認するには、まず、署名を復号してハッシュ値を得る。次に、アプリケーション本体に対してハッシュ関数を適用し、得られる値が、署名を復号して得られるハッシュ値と同じであるかどうかを確認する。あるいは、アプリケーション本体にハッシュ関数を適用してハッシュ値を得て、それに対して暗号化を行い、得られたものが署名と同じであるかどうかを確認する。前者の方法は、例えば、署名が公開鍵暗号化方式を用いる場合に普通に利用される方法であり、署名は、ハッシュ値を署名を行なう者の秘密鍵によって暗号化され、アプリケーション本体が改ざんされていないことを確認する時に、署名を行なった者の秘密鍵に対応する公開鍵によって復号される。後者の方法は、例えば、アプリケーション本体が改ざんされていないことを確認する者が署名を行なった者の秘密鍵を知っている場合や、共通鍵暗号化方式が用いられる場合に使用される。

10

【 0 0 4 6 】

本実施の形態において、T R M アクセスライブラリ部 4 0 1 は、自身を認証モジュールに認証させた後で、ダウンロード部 1 0 2 にダウンロードされたアプリケーションの認証のための処理の一部である処理、すなわち、ダウンロード部 1 0 2 にダウンロードされたアプリケーションから署名認証用ダイジェストを生成することを行なう。「署名認証用ダイジェスト」とは、上に挙げた S H A - 1 や M D 5 などのハッシュ関数によるハッシュ値である。

20

【 0 0 4 7 】

署名認証情報出力部 5 0 1 は、T R M アクセスライブラリ部 4 0 1 で生成された署名認証用ダイジェスト 5 0 4 と署名 5 0 5 とを含む署名認証情報 5 0 6 を認証モジュールに出力する。ここに「署名 5 0 5」とは、ダウンロード部 1 0 2 にダウンロードされたアプリケーションに付加された署名である。

【 0 0 4 8 】

署名認証情報入力部 5 0 2 は、署名認証情報出力部 5 0 1 から出力された署名認証情報 5 0 6 を入力する。

【 0 0 4 9 】

署名認証部 5 0 3 は、署名認証情報入力部 5 0 2 から入力される署名認証用ダイジェストと署名とに基づいて署名の検証を行なう。検証の方法としては、例えば、署名が公開鍵暗号化方式の秘密鍵で暗号化されている場合には、その秘密鍵に対応する公開鍵で復号し、その復号の結果と署名認証用ダイジェストとを比較して等しいかどうかで認証を行なう。あるいは、耐タンパ領域に共通鍵を保持しておき、その共通鍵で署名を復号して署名認証用ダイジェストと比較する方法、あるいは、その秘密鍵を耐タンパ領域に保持して、その秘密鍵で署名認証用ダイジェストを暗号化して署名と比較する方法がある。

30

【 0 0 5 0 】

図 7 は、端末 1 0 0 の動作を説明するフローチャートである。このフローチャートの処理を行なう前提として、T R M アクセスライブラリ部 4 0 1 は、認証モジュールにより認証されているとする。ステップ S 7 0 1 において、ダウンロード部 1 0 2 にアプリケーションをダウンロードする。ステップ S 7 0 2 において、ダウンロードしたアプリケーションより署名認証用ダイジェストを、T R M アクセスライブラリ部 4 0 1 で生成する。ステップ S 7 0 3 において、ステップ S 7 0 2 で求められた署名認証用ダイジェストとダウンロードされたアプリケーションに付加された署名とを含む署名認証情報 5 0 6 を、署名認証情報出力部 5 0 1 より認証モジュール 1 0 1 に出力する。

40

【 0 0 5 1 】

図 8 は、認証モジュール 1 0 1 の署名認証情報入力部 5 0 2 と署名認証部 5 0 3 との動作を説明するフローチャートである。このフローチャートの処理を行なう前提として、T R M アクセスライブラリ部 4 0 1 は、認証モジュールにより認証されているとする。このために、例えば、T R M アクセスライブラリ部 4 0 2 は認証の結果を認証モジュール

50

101内に設定しておくものとし、図8のフローチャートの処理を行なうときには、その設定された認証の結果を参照して、TRMアクセスライブラリ部401が認証されている場合に限り、図8のフローチャートの処理を行なうようにする方法がある。ステップS801において、署名認証情報506を、署名認証情報入力部502により入力する。ステップS802において、署名認証用ダイジェスト504と署名505とに基づいて署名の検証を行なう。署名の検証の方法については、上で説明した通りである。

【0052】

本実施の形態においては、TRMアクセスライブラリ部401は、認証モジュール101によって認証がされたことを条件にして、ダウンロード部102にダウンロードされたアプリケーションの署名認証用ダイジェストを生成し、生成された署名用認証用ダイジェストが認証モジュール101に入力されるので、署名認証用ダイジェストは信頼のおけるものであり、ダウンロード部102にダウンロードされたアプリケーションの認証を認証モジュール101が行なうことができる。

【0053】

(実施の形態3)

図9は、本発明の実施の形態3にかかわる認証モジュール101の機能ブロック図を示す。本実施の形態は、実施の形態2における署名認証情報による認証方法をより具体的にしたものであり、実施の形態2の認証モジュール101が、署名由来ダイジェスト生成情報取得部901と、署名由来ダイジェスト生成部902と、をさらに有している。

【0054】

署名由来ダイジェスト生成情報取得部901は、署名認証情報入力部502から入力される署名を利用して署名由来ダイジェストを生成するための署名由来ダイジェスト生成情報を取得する。署名が、アプリケーション本体のハッシュ値を暗号化したものであれば、「署名由来ダイジェスト生成情報」は、暗号化したものを復号する復号鍵である。暗号化として公開鍵暗号化方式が使用されている場合には、アプリケーション本体のハッシュ値を暗号化するために用いられた秘密鍵に対応する公開鍵が署名由来ダイジェスト生成情報となる。この公開鍵は、認証モジュールに保持されていてもよい。また、端末100などを經由して適当なサーバより取得するようになっていてもよい。

【0055】

署名由来ダイジェスト生成部902は、署名認証情報入力部から入力された署名903と、署名由来ダイジェスト生成情報取得部に保持された署名由来ダイジェスト生成情報と、を利用して署名由来ダイジェスト905を生成する。すなわち、署名由来ダイジェスト生成情報が公開鍵であれば、署名903をその公開鍵で復号してアプリケーション本体のハッシュ値である署名由来ダイジェスト905を生成する。

【0056】

署名認証部503は、署名由来ダイジェスト生成部902で生成された署名由来ダイジェスト905と、署名認証情報入力部502から入力された署名認証用ダイジェスト904と、に基づいて認証を行なう。すなわち、署名由来ダイジェスト905と署名認証用ダイジェスト904とを比較し、同じものであれば、ダウンロード部102にダウンロードされたアプリケーションを認証し、異なるものであれば、認証しない。

【0057】

図10は、本実施の形態における署名認証情報入力部502と、署名由来ダイジェスト生成情報取得部901と、署名由来ダイジェスト生成部902と、署名認証部503との動作を説明するフローチャートである。このフローチャートの処理を行なう前提として、TRMアクセスライブラリ部401は、認証モジュール101により認証されているとする。このためには例えば、TRMアクセスライブラリ部認証部402は認証の結果を認証モジュール101内に設定しておくものとし、図10のフローチャートの処理を行なうときには、その設定された認証の結果を参照して、TRMアクセスライブラリ部401が認証されている場合に限り、図10のフローチャートの処理を行なうようにする方法がある。ステップS1001において、署名認証情報506を、署名認証情報入力部502により

10

20

30

40

50

入力し、署名903と署名認証用ダイジェスト904を得る。ステップS1002において、署名由来ダイジェスト生成情報を、署名由来ダイジェスト生成情報取得部901により取得する。ステップS1003において、署名903と署名由来ダイジェスト生成情報から、署名由来ダイジェスト生成部902において、署名由来ダイジェスト905を生成する。ステップS1004において、署名由来ダイジェスト905と署名認証用ダイジェスト904に基づいて、署名認証部503において認証を行なう。

【0058】

(実施の形態4)

図11は、本発明の実施の形態4に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態においては、実施の形態2または実施の形態3におけるアプリケーション認証システムの端末がさらに認証モジュール認証部1101を有している。

【0059】

認証モジュール認証部1101は、認証モジュール101を認証する。この認証の方法としては、図12に示したフローチャートに示した方法がある。この方法を用いるにあたっては、認証モジュールに対して公開鍵暗号化方式の秘密鍵とそれに対応する公開鍵が生成され、認証モジュールは、その秘密鍵を格納していることを前提とする。ステップS1201において、認証モジュール認証部1101は、乱数を発生させる。ステップS1202において、認証モジュール101の持つ公開鍵によりステップS1201で発生させた乱数を暗号化する。ステップS1203において、認証モジュール101に、ステップS1202で暗号化した乱数を入力し、復号するように要求する。ステップS1204において、認証モジュール101からの復号の結果を受け取り、ステップS1205において、ステップS1201で発生させた乱数とステップS1204で受け取った復号の結果が等しいか判断する。別の方法としては、認証モジュール認証部1101は、発生させた乱数を認証モジュール101に入力し、その乱数を認証モジュール101の秘密鍵で暗号化するように要求する方法がある。認証モジュール認証部1101は、その暗号化の結果を得て、認証モジュール101の公開鍵で復号し、認証モジュール101に入力した乱数と等しいかどうかを判断する。

【0060】

このように、端末100が認証モジュール101を認証するための認証モジュール認証部1101を有することにより、端末100が認証モジュール101を認証することができ、端末100内で動作するアプリケーションが機密性の高い情報(プライバシー情報、E-コマース(EC)のログ、銀行取引における口座残高など)を認証モジュール101に書き込む際に、認証モジュール101が正当なものであるかどうかを確認することができる。

【0061】

(実施の形態5)

図13は、本発明の実施の形態5に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態は、実施の形態4におけるアプリケーション認証システムのTRMアクセスライブラリ部401が、アプリケーション利用リソース情報保持手段1301をさらに有している形態である。

【0062】

アプリケーション利用リソース情報保持手段1301は、アプリケーション利用リソース情報を保持する。「アプリケーション利用リソース情報」とは、認証されたアプリケーションに対して利用を認めるリソースに関する情報である。「認証されたアプリケーション」とは、ダウンロード部102にダウンロードされたアプリケーションであって、TRMアクセスライブラリ部401による認証のための処理が行なわれ、正しく認証されたアプリケーションである。「リソース」とは、アプリケーションが利用するアプリケーションの外部の資源である。リソースには、端末100およびそれに装着された認証モジュール101の資源であるローカルリソースと、それ以外のリソース、例えば、端末100の通信先であるサーバの資源であるリソースがある。ローカルリソースには、メモリの使用、

ファイルの使用、I r D Aの使用、B l u e t o o t hの使用、通信の使用、T R Mの使用、アプリケーションの使用、非接触/接触I CカードI / Fの使用などの種類がある。また、ローカルリソースのうち、メモリの使用などの場合には、使用できるメモリの量やメモリの番地などの範囲に関するものもある。また、使用できる時間という範囲も挙げることができる。

【 0 0 6 3 】

図14は、アプリケーション利用リソース情報を例示している。図14において、アプリケーション利用リソース情報1400は、メモリの使用1401、ファイルの使用1402、I r D Aの使用1403、B l u e t o o t hの使用1404、通信の使用1405、T R Mの使用1406、アプリケーションの使用1407、非接触/接触I CカードI / Fの使用1408、動作1409、使用日時1410などの項目から成っている。

10

【 0 0 6 4 】

メモリの使用1401の項目としては、例えば、メモリとして使用可能な量、使用可能な範囲のアドレス、書き込みができる回数、読み出しができる回数、メモリが使用可能な日時、使用不可能な日時などを挙げるることができる。フラッシュメモリにとっては、書き込みは負担のかかる操作であるので、メモリの書き込みの回数を制限することには特に意味がある。

【 0 0 6 5 】

ファイルの使用1402の項目としては、端末100の持つファイルや認証モジュール101などの端末100に接続された外部のメモリの持つファイルに対するアクセスの制限を記述するものが挙げられ、例えば、アクセス可能なディレクトリ名、アクセス可能なファイル名、アクセス可能なファイルの種類(例えば、ファイルの拡張子によって指定される)、使用可能な日時、使用不可能な日時などがある。

20

【 0 0 6 6 】

I r D Aの使用1403は、端末100の持つ赤外線通信の機能の使用を認めるかどうかを表す項目であり、その機能が使用可能/不可能な時刻、使用可能なトータル時間、使用回数などを挙げるることができる。

【 0 0 6 7 】

B l u e t o o t hの使用1404は、端末100の持つB l u e t o o t hによる通信の機能の使用を認めるかどうかを表す項目であり、その機能が使用可能な時刻、使用可能なトータル時間、使用回数のほかに、使用可能な電波の強度、ローミング回数、使用可能/不可能な日時、などを挙げるることができる。使用可能な電波の強度を指定することにより、通信可能な距離を指定することができる。

30

【 0 0 6 8 】

通信の使用1405は、サーバなどとの通信の機能の使用を認めるかどうかを表す項目であり、その機能が使用可能/不可能な時刻、使用可能なトータル時間、使用回数、使用可能な電波の強度、ローミング回数のほかに、アクセス可能なサーバなどを挙げることができる。アクセス可能なサーバは、I Pアドレスによるアドレスの指定、ドメイン名による指定、あるいは、メールサーバやF T Pサーバなどのサーバの機能あるいはサーバとの通信のプロトコルによる指定により、アプリケーションがサーバなどを利用できるかどうか

40

【 0 0 6 9 】

T R Mの使用1406は、認証モジュール101などの耐タンパ領域へのアクセスを認めるかどうかを表す項目であり、その耐タンパ領域へのアクセスができる日時、アクセスができない日時、アクセス回数や、耐タンパ領域を持つI Cカード内で動作するカードアプリケーションであって、アクセス可能なカードアプリケーション、利用可能なI Cカードコマンドの種類などが挙げられる。

【 0 0 7 0 】

アプリケーションの使用1407は、アプリケーションが連携可能な他のアプリケーションを指定する項目である。例えば、アドレス帳やメール、スケジューラ、ゲームなどであ

50

る。また、他のアプリケーションと連携することが可能な日時も含まれていてもよい。端末100が携帯電話である場合には、電話通信時にアプリケーションが通話と並行して動作することを認めるかどうか、通話の開始時に停止あるいは終了しなければならないかどうかを指定するようにしてもよい。

【0071】

接触／非接触ICカードI/Fの使用1408は、端末100と通信が可能な接触／非接触ICカードやICカードリーダーと通信するためのインターフェースの使用を認めるかどうかの項目である。そのインターフェースが使用可能な時刻、使用可能なトータルの時間、使用可能／不可能な回数、使用可能I/F (Type A、Type B、Type Cなど)、利用可能なICカードコマンドの種類などが挙げられる。

10

【0072】

使用日時1410は、アプリケーションが動作可能な日時を指定する。あるいは、アプリケーションの動作を停止すべき日時を指定する。

【0073】

ダウンロード部102にダウンロードされたアプリケーションが、リソースを使用する際には、TRMアクセスライブラリ部401に対してリソースに使用の要求を出し、TRMアクセスライブラリ部401は、アプリケーション利用リソース情報保持手段1301に保持されているアプリケーション利用リソース情報を参照し、要求されたリソースが使用可能かどうかを判断し、もし、使用可能であれば、その要求されたリソースをアプリケーションに使用させる。

20

【0074】

アプリケーション利用リソース情報は、ダウンロード部102にダウンロードされるアプリケーションとともにダウンロードされて、アプリケーション利用リソース情報保持手段1301に保持されるようになっていてもよい。図15は、アプリケーションとともにアプリケーション利用リソース情報がダウンロードされるデータを模式的に表現した図である。まず、アプリケーション本体であるアプリケーションデータ1501があり、アプリケーションデータ1501を認証するための署名であるアプリケーションデータ署名データ1502が続き、その後ろに、アプリケーション利用リソース情報1503があり、アプリケーション利用リソース情報を認証するためのアプリケーション利用リソース情報署名データ1504がある。アプリケーション利用リソース情報1503には、符号1505が付された部分のように、「IrDA 1」により、IrDAが使用可能であり、「Bluetooth 0」により、Bluetoothは使用不可、であることなどが表現される。

30

【0075】

また、アプリケーション利用リソース情報は、認証モジュール101に格納され、必要に応じて読み出され、アプリケーション利用リソース情報保持手段1301により保持されるようになっていてもよい。

【0076】

このように、アプリケーション利用リソース情報保持手段1301をTRMアクセスライブラリ部401が有する構成により、ダウンロードされたアプリケーションが使用可能な資源を制限することが可能となる。これにより、アプリケーション利用リソース情報をアプリケーションの製作者やサービス提供者などに対して発行して対価を得るというビジネスを行なうことができる。このローカルリソースの許可の制御には、アプリケーション利用リソース情報を用いることができ、特定のアプリケーションに対してローカルリソースの利用を細かく許可／不許可にすることができる。ローカルリソースの許可を行なう場合には、アプリケーション利用リソース情報の発行者に使用料を支払うようにすれば、アプリケーション利用リソース情報の発行による商取引が可能となる。また、端末100の利用者は、対価を支払うことにより、端末100にダウンロードされたアプリケーションのリソースの使用に対する制限がより少ないアプリケーション利用リソース情報を入手することが可能となり、端末100の利用者を相手とする商取引も実現できる。

40

50

【 0 0 7 7 】

なお、実施の形態 2 ないし実施の形態 5 において、ダウンロード部 1 0 2 が使用許諾書をダウンロードしてもよい。「使用許諾書」とは、ダウンロードアプリケーションの署名付きのアプリケーション利用リソース情報である。「ダウンロードアプリケーション」とは、ダウンロード部 1 0 2 にダウンロードされたアプリケーションである。「署名付きのアプリケーション利用リソース情報」とは、アプリケーション利用リソース情報にそのアプリケーション利用リソース情報の署名を付したものである。アプリケーション利用リソース情報は、ダウンロードされたアプリケーションが端末 1 0 0 や認証モジュール 1 0 1 のリソースを使用するための許可証であるので、アプリケーション利用リソース情報の真正性を保証することは重要であり、そのために署名をアプリケーション利用リソース情報に付けるようにする。

10

【 0 0 7 8 】

なお、ダウンロードされたアプリケーションの署名の検証と、アプリケーション利用リソース情報の署名の検証とは、同時に行なわれても、異なる時に行なわれてもよい。例えば、最初にアプリケーションの署名の検証が行なわれ、アプリケーションが動作し、リソースにアクセスをしたときに、アプリケーション利用リソース情報の署名の検証が行われ、真正なものであることを確認してリソースのアクセスが許されるかどうかを判断するようにしてもよい。また、ダウンロードされたアプリケーションの署名とアプリケーション利用リソース情報の署名の作成者は、同じであっても異なってもよい。署名の作成者が異なることがあり得る理由は、アプリケーションの作成者とアプリケーション利用リソース情報の発行者とは異なり、前者が後者に対してリソースの利用の許可を申請して、後者よりアプリケーション利用リソース情報の発行を受けることがあるからである。なお、リソースの利用の許可の申請の際に、対価の収受が行なわれてもよい。

20

【 0 0 7 9 】

また、使用許諾書のアプリケーション利用リソース情報に、アプリケーションが、リソースにアクセスすることができる時間的限度を示す期限日情報が含まれている場合には、その期限日情報に基づいて認められる時間的情報がすでに期限切れである場合には、ダウンロード部 1 0 2 は、使用許諾書をダウンロードし、使用許諾書を更新するようにしてもよい。

【 0 0 8 0 】

また、ダウンロードされたアプリケーションの実行時、又は / 及び、アプリケーションの認証時に使用許諾書をダウンロードするようにしてもよい。使用許諾書のダウンロードは、端末 1 0 0 が通信できるサーバより行なってもよい。また、認証モジュール 1 0 1 より行なってもよい。

30

【 0 0 8 1 】

また、ダウンロード部 1 0 2 は、サーバよりダウンロードした使用許諾書を保持し続けることも考えられるが、この場合、使用許諾書がサーバ内において更新されている場合があり、ダウンロード部 1 0 2 で保持されている使用許諾書が失効している可能性がある。そこで、ダウンロード部 1 0 2 は、ダウンロードされたアプリケーションの実行時、又は / 及び、アプリケーションの認証時に、ダウンロードされた使用許諾書の有効性をサーバに問い合わせるようにしてもよい。

40

【 0 0 8 2 】

また、ダウンロード部 1 0 2 は、ダウンロード部 1 0 2 あるいは端末 1 0 0 の他の部が、使用許諾書の内容をサーバに対してオンラインにて問い合わせ、アプリケーションのリソースの利用が許可されているかどうかを問い合わせるようになっていてもよい。

【 0 0 8 3 】

(実施の形態 6)

図 1 6 は、本発明の実施の形態 6 に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態は、実施の形態 4 または実施の形態 5 のアプリケーション認証システムの端末 1 0 0 の T R M アクセスライブラリ部 4 0 1 がアプリケーション利用リソー

50

ス情報出力手段１６０１を有する。

【００８４】

アプリケーション利用リソース情報出力手段１６０１は、認証モジュール認証部１１０１による認証がされた認証モジュールに対してアプリケーション利用リソース情報を出力する。

【００８５】

本実施の形態において、認証モジュール１０１のＴＲＭ部１０３は、アプリケーション利用リソース情報出力手段１６０１から出力されたアプリケーション利用リソース情報を耐タンパ領域に書き換え可能に保持することを行なう。

【００８６】

このように耐タンパ領域に保持されたアプリケーション利用リソース情報は、必要に応じて端末１００に読み込まれ、ダウンロード部１０２にダウンロードされたアプリケーションが資源を使用しようとする際に、その資源が使用できるかどうかを判断するために参照される。

【００８７】

このように、端末１００のＴＲＭアクセスライブラリ部４０１がアプリケーション利用リソース情報出力手段１６０１を有し、アプリケーション利用リソース情報出力手段１６０１により出力されたアプリケーション利用リソース情報をＴＲＭ部１０３が保持することにより、認証モジュール１０１に保持された状態でアプリケーション利用リソース情報が提供された後でも、必要に応じてアプリケーション利用リソース情報を書き換えることができる。例えば、アプリケーション利用リソース情報の有効期限を書き換えることなどが可能となる。また、例えば、サービスの提供者やサービスの利用者が対価を支払うことにより、耐タンパ領域が保持するアプリケーション利用リソース情報を、アプリケーションの使用可能な資源を制限することが少ないものに更新することができる。書き換えは、あらかじめ署名認証情報入力部５０２によって認証されたＴＲＭアクセスライブラリ部４０１により行なわれるので、不正な書き換えを防止することができる。

【００８８】

（実施の形態７）

図１７は、本発明の実施の形態７に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態は、実施の形態５または実施の形態６におけるアプリケーション認証システムの端末１００がアプリケーション利用リソース情報ダウンロード部１７０１を有している。

【００８９】

アプリケーション利用リソース情報ダウンロード部１７０１は、署名１７０３が付されたアプリケーション利用リソース情報１７０２をダウンロードする。このダウンロードは、図１５に示すようにダウンロード部１０２にダウンロードされるアプリケーションとともにダウンロードされてもよい。また、ダウンロード部１０２にダウンロードされたアプリケーションとは別にダウンロードされてもよい。例えば、先にアプリケーションをダウンロードしておき、そのアプリケーションが資源にアクセスしようとするときに、アプリケーション利用リソース情報がダウンロードされるようになっていてもよい。

【００９０】

本実施の形態において、ＴＲＭアクセスライブラリ部４０１が、アプリケーション利用リソース情報ダウンロード部１７０１にダウンロードされたアプリケーション利用リソース情報１７０２に付された署名１７０３を認証するようにしてもよい。ＴＲＭアクセスライブラリ部４０１は、認証モジュール１０１によって認証されるので、正しく認証されたＴＲＭアクセスライブラリ部４０１により、アプリケーション利用リソース情報１７０２の署名１７０３を認証した結果は認証モジュール１０１にとって信頼できるものになっている。したがって、このように認証されたアプリケーション利用リソース情報１７０２に従って、ダウンロードされたアプリケーションが認証モジュール１０１へアクセスを許しても不正な操作は発生しないことが保証される。

10

20

30

40

50

【 0 0 9 1 】

(実施の形態 8)

図 1 8 は、本発明の実施の形態 8 に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態は、実施の形態 5 または実施の形態 6 におけるアプリケーション認証システムの端末 1 0 0 がアプリケーション利用リソース情報ダウンロード部 1 7 0 1 と、アプリケーション利用リソース情報署名認証情報出力部 1 8 0 1 を有し、認証モジュール 1 0 1 がアプリケーション利用リソース情報署名認証情報入力部 1 8 0 2 と、アプリケーション利用リソース情報署名認証部 1 8 0 3 とを有している。本実施の形態は、実施の形態 7 において、T R M アクセスライブラリ部 4 0 1 がアプリケーション利用リソース情報 1 7 0 2 に付された署名 1 7 0 3 を認証するかわりに、認証モジュールによって署名 1 7 0 3 を認証する形態である。

10

【 0 0 9 2 】

アプリケーション利用リソース情報ダウンロード部 1 7 0 1 は、署名 1 7 0 3 が付されたアプリケーション利用リソース情報 1 7 0 2 をダウンロードする。

【 0 0 9 3 】

本実施の形態において、T R M アクセスライブラリ部 4 0 1 は、アプリケーション利用リソース情報ダウンロード部 1 7 0 1 にダウンロードされたアプリケーション利用リソース情報 1 7 0 2 から署名認証用ダイジェストを生成し、アプリケーション利用リソース情報署名認証情報出力部 1 8 0 1 は、生成された署名認証用ダイジェストと、署名 1 7 0 3 とを含む署名認証情報 1 8 0 6 を認証モジュールに出力する。

20

【 0 0 9 4 】

アプリケーション利用リソース情報署名認証情報入力部 1 8 0 2 は、アプリケーション利用リソース情報署名認証情報出力部 1 8 0 1 から出力された署名認証情報 1 8 0 6 を入力する。署名認証情報 1 8 0 6 は、T R M アクセスライブラリ部 4 0 1 で生成された署名認証用ダイジェスト 1 8 0 4 と、アプリケーション利用リソース情報 1 7 0 2 に付された署名 1 7 0 3 である署名 1 8 0 5 と、を含む。

【 0 0 9 5 】

アプリケーション利用リソース情報署名認証部 1 8 0 3 は、アプリケーション利用リソース情報署名認証情報入力部 1 8 0 2 から入力される署名認証用ダイジェスト 1 8 0 4 と署名 1 8 0 5 とに基づいて署名の検証を行なう。

30

【 0 0 9 6 】

図 1 9 は、本実施の形態における端末 1 0 0 の動作を説明するフローチャートである。ステップ S 1 9 0 1 において、アプリケーション利用リソース情報 1 7 0 2 より、T R M アクセスライブラリ部 4 0 1 にて、署名認証用ダイジェスト 1 8 0 4 を生成する。ステップ S 1 9 0 2 において、署名認証用ダイジェスト 1 8 0 4 と署名 1 8 0 5 とを含む署名認証情報 1 8 0 6 を、アプリケーション利用リソース情報署名認証情報出力部 1 8 0 1 により、認証モジュール 1 0 1 に出力する。ステップ S 1 9 0 3 において、認証モジュール 1 0 1 より認証結果を受け取ることを行なう。

【 0 0 9 7 】

図 2 0 は、本実施の形態における認証モジュール 1 0 1 の動作を説明するフローチャートである。ステップ S 2 0 0 1 において、署名認証情報 1 8 0 6 を、アプリケーション利用リソース情報署名認証情報入力部 1 8 0 2 により入力する。ステップ S 2 0 0 2 において、署名認証用ダイジェスト 1 8 0 4 と署名 1 8 0 5 とに基づいて署名 1 8 0 5 の署名の検証を、アプリケーション利用リソース情報署名認証部 1 8 0 3 において行なう。ステップ S 2 0 0 3 において、検証の結果を端末 1 0 0 へ返す。

40

【 0 0 9 8 】

このような実施の形態において、認証モジュール 1 0 1 によって認証された T R M アクセスライブラリ部 4 0 1 により生成された署名用認証用ダイジェストに基づいてアプリケーション利用リソース情報 1 7 0 2 の署名が認証されるので、その認証の結果は信頼できるものになる。また、アプリケーション利用リソース情報署名認証部 1 8 0 3 での認証を、

50

暗号化に基づかずに、認証モジュール101のTRM部に格納された署名に適合するかどうかで認証することも可能になるので、認証を簡単な手間で行なうことができる。

【0099】

(実施の形態9)

本発明の実施の形態9は、認証モジュール内で動作する認証モジュール内アプリケーションが、TRMアクセスライブラリ部が認証モジュールによって認証されていることを条件に、端末内で動作するアプリケーションからのアクセスを受け入れることを特徴とする。

【0100】

図21は、本実施の形態に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態は、実施の形態2ないし実施の形態3におけるアプリケーション認証システムの端末100が端末アプリケーション保持部2101を有し、認証モジュール101は、認証モジュール内アプリケーション保持部2103を備えるTRM部103を有する。

10

【0101】

端末アプリケーション保持部2101は、認証モジュール101のTRM部にアクセスをする端末アプリケーション2102を保持する。「端末アプリケーション2102」とは、端末100の内部で実行されるアプリケーションである。そのアプリケーションは、ダウンロード部102によってダウンロードされたアプリケーションであってもよく、また、端末のROMに保持されているアプリケーションであってもよい。「保持する」とは、端末アプリケーション2102を実行可能にすることである。したがって、端末アプリケーション保持部2101は、端末アプリケーション2102を実行させるために、端末アプリケーション2102の全部または一部をロードする端末100の書き換え可能なメモリ領域により実現される。

20

【0102】

認証モジュール内アプリケーション保持部2103は、認証モジュール内アプリケーション2104を保持する。「認証モジュール内アプリケーション2104」とは、認証モジュール101内で動作するアプリケーションである。認証モジュール101がICカードであれば、認証モジュール内アプリケーション2104はカードアプリケーションとなる。「保持する」とは、認証モジュール内アプリケーション2104を実行可能にすることである。したがって、認証モジュール内アプリケーション保持部2103は、認証モジュール内アプリケーション2104を実行させるために、認証モジュール内アプリケーション2104の全部または一部をロードする認証モジュールの書き換え可能なメモリ領域により実現される。

30

【0103】

本実施の形態において、認証モジュール内アプリケーション2104は、TRMアクセスライブラリ部認証部402によるTRMアクセスライブラリ部401の認証を条件として端末アプリケーション2102からのアクセスを受け入れて動作する。このために、認証モジュール内にTRMアクセスライブラリ部401が認証されたかどうかを示す値を格納しておき、図22に示すように、ステップS2201において、その値自身あるいは、そのような値が存在するかどうかを調べてTRMアクセスライブラリ部401の認証が成功したかどうかを判定する。もし、認証が成功していれば、ステップS2202に移行し、端末アプリケーション2102からのアクセスを受け入れる。ステップS2201の判定は、認証モジュール内アプリケーション2104で行なう場合と、認証モジュール内アプリケーション2104以外において行なう場合がある。認証モジュール内アプリケーション2104で行なう場合は、認証モジュール内アプリケーション2104が起動してから、端末アプリケーション2102がアクセスするまでの間に、TRMアクセスライブラリ部401が認証されたかどうかを示す値を見て、判断することになる。認証モジュール内アプリケーション2104以外においてステップS2201の判定を行なう場合は、認証モジュール内アプリケーション2104が起動される際に、TRMアクセスライブラリ部401が認証されたかどうかを示す値が確認される。

40

50

【 0 1 0 4 】

認証モジュール 1 0 1 が I C カードである場合には、認証モジュール内アプリケーション 2 1 0 4 を起動するのは、カードマネージャであるので、カードマネージャが認証モジュール内アプリケーション 2 1 0 4 を起動するかどうかを、T R M アクセスライブラリ部 4 0 1 が認証されたかどうかを示す値を見て判断することになる。また、認証モジュール 1 0 1 の端末とのインターフェース部分（図 2 1 において図示せず）が、端末アプリケーション 2 1 0 2 から認証モジュール内アプリケーション 2 1 0 4 へのアクセスを許可するかどうかを決める際に、T R M アクセスライブラリ部 4 0 1 が認証されたかどうかを示す値を見て判断するようにしてもよい。

【 0 1 0 5 】

また、認証モジュール内アプリケーション 2 1 0 4 が、端末アプリケーション 2 1 0 2 からのアクセスを受け付けるたびに、T R M アクセスライブラリ部 4 0 1 が認証されたかどうかを示す値を見て、そのアクセスを受け入れて動作するかどうかを決定するようにしてもよい。図 2 3 は、このような場合の認証モジュール内アプリケーションの動作を説明するフローチャートである。ステップ S 2 3 0 1 において、端末アプリケーション 2 1 0 2 からのアクセスを、認証モジュール内アプリケーション 2 1 0 4 で受け付ける。ステップ S 2 3 0 2 において、T R M アクセスライブラリ部 4 0 1 が認証されたかどうかを示す値を、認証モジュール内アプリケーション 2 1 0 4 がチェックして、T R M アクセスライブラリ部 4 0 1 の認証が成功したかどうかを判断し、もし、その認証が成功していれば、ステップ S 2 3 0 3 において、端末アプリケーション 2 1 0 2 からのアクセスを受け入れて動作する。

【 0 1 0 6 】

図 2 4 は、T R M アクセスライブラリ部 4 0 1 が認証されたかどうかを示す値を T R M 部 1 0 3 に格納するようにした形態を示す。T R M 部 1 0 3 は、認証結果識別子生成手段 2 4 0 1 を有している。認証結果識別子生成手段 2 4 0 1 は、T R M アクセスライブラリ部 4 0 1 による T R M アクセスライブラリ部 4 0 2 による T R M アクセスライブラリ部 4 0 1 の認証の成功を条件として認証結果識別子 2 4 0 2 を生成する。認証モジュール内アプリケーション 2 1 0 4 は、認証結果識別子 2 4 0 2 の存在を条件として、端末アプリケーション 2 1 0 2 からのアクセスを受け入れる。

【 0 1 0 7 】

なお、認証結果識別子 2 4 0 2 は、T R M アクセスライブラリ部 4 0 2 による T R M アクセスライブラリ部 4 0 1 の認証の成功を示すだけでなく、認証の失敗を示す内容を持つものであってもよい。この場合には、認証結果識別子生成手段 2 4 0 1 は、T R M アクセスライブラリ部 4 0 2 による T R M アクセスライブラリ部 4 0 1 の認証の成功 / 失敗に応じた内容を持つ認証結果識別子 2 4 0 2 を生成する。また、認証モジュール内アプリケーション 2 1 0 4 は、認証結果識別子 2 4 0 2 の内容を見て、認証が成功したかどうかを判断する。

【 0 1 0 8 】

図 2 5 は、認証モジュール 1 0 1 が I C カードである場合における、認証結果識別子の実現方法を示す。図 2 5 において、T R M アクセスライブラリ部 4 0 1 を認証するのがカードアプリケーション A (2 5 0 1) であり、T R M アクセスライブラリ部 4 0 1 が認証されると、書き換えが可能なメモリ領域、例えば、R A M 領域 2 5 0 3 に認証結果識別子を設定する。図 2 5 において、旗の印が認証結果識別子を模式的に表現している。R A M 領域 2 5 0 3 は、カードアプリケーション A (2 5 0 1) は読み書きすることができるが、I C カード内のアプリケーション間で、それぞれ悪影響を及ぼし合わないよう、各アプリケーションを独立して起動させるためのファイアウォール機能により、別のカードアプリケーション B (2 5 0 2) は、R A M 領域 2 5 0 3 に直接アクセスすることができない。そこで、カードアプリケーション A (2 5 0 1) が相手先を指定してインターフェースを公開できる公開インターフェース (S h a r a b l e I n t e r f a c e) 機能を利用して公開インターフェース 2 5 0 4 を提供し、この公開インターフェース 2 5 0 4 を通し

10

20

30

40

50

て、端末アプリケーション 2 4 0 2 からアクセスがされたカードアプリケーション 2 5 0 2 は、R A M 領域 2 5 0 3 に認証結果識別子が存在するかどうかを確認する。

【 0 1 0 9 】

また、図 2 6 は、認証モジュール 1 0 1 が I C カードである場合における、認証結果識別子の別の実現方法を示す。符号 2 6 0 1、2 6 0 2 が付された矩形はデディケートィッドファイル (D F) を表している。各 D F は各カードアプリケーションに対応している。このため、D F がセレクトされると対応するカードアプリケーションが起動する。以後においては、符号 2 6 0 1 の D F は、カードアプリケーション A に対応し、同様に符号 2 6 0 2 の D F は、アプリケーション B に対応しているとする。符号 2 6 0 3、2 6 0 4、2 6 0 5、2 6 0 6、2 6 0 7 が付された矩形は、エレメンタリファイル (E F) を表している。D F に対応するカードアプリケーションが起動すると、それに対応する D F の直下の E F が操作できるようになっている。例えば、符号 2 6 0 1 の D F がセレクトされ、カードアプリケーション A が起動すると、カードアプリケーション A は、符号 2 6 0 3、2 6 0 4 の E F をアクセスすることができるようになる。

10

【 0 1 1 0 】

以降では、カードアプリケーション A が T R M アクセスライブラリ部 4 0 1 を認証し、正しく認証された場合に、符号 2 6 0 4 の E F に認証結果識別子をセットすると仮定する。符号 2 6 0 4 が付された E F の状態を符号 2 6 0 1 が付された D F のセキュリティステータスに含めることにより、D F と E F とが形成する木構造における符号 2 6 0 1 が付された D F の子孫に相当する D F のセレクトを制御することができる。つまり、符号 2 6 0 2 の D F のセレクトは、符号 2 6 0 1 が付された D F の子孫のいずれか、ここでは、符号 2 6 0 4 が付された E F 内の識別子の存在を条件とする設定をする。すなわち、カードアプリケーション A による T R M アクセスライブラリ部 4 0 1 の認証の結果により、カードアプリケーション B に対応する D F のセレクトを制御することができるようになるので、カードアプリケーション B を T R M アクセスライブラリ部 4 0 1 が認証された場合にのみ起動させることができるようになる。

20

【 0 1 1 1 】

このような実施の形態により、T R M アクセスライブラリ部 4 0 1 が認証モジュール 1 0 1 によって認証されない限り、認証モジュール内アプリケーション 2 1 0 4 は端末アプリケーション 2 1 0 2 からアクセスされることがないので、認証モジュール 1 0 1 のセキュリティが保たれる。

30

【 0 1 1 2 】

なお、認証モジュール内アプリケーション保持部 2 1 0 3 は、T R M 部 1 0 3 の外部にあってもよい。その場合、認証モジュール内アプリケーション 2 1 0 4 は、T R M アクセスライブラリ部 4 0 2 による T R M アクセスライブラリ部 4 0 1 が認証されているかどうかを調べて端末アプリケーション 2 1 0 2 のアクセスを受け入れて動作することになる。

【 0 1 1 3 】

(実施の形態 1 0)

図 2 7 は、本発明の実施の形態 1 0 に関するアプリケーション認証システムの機能ブロック図を示す。本発明の実施の形態 1 0 では、認証モジュール内アプリケーションは、端末で動作するアプリケーションの認証を条件として、そのアプリケーションからのアクセスを受け入れて動作する。図 2 7 は、本実施の形態におけるアプリケーション認証システムの機能ブロック図を示す。本実施の形態におけるアプリケーション認証システムにおいては、実施の形態 9 におけるアプリケーション認証システムの T R M 部 1 0 3 が、アプリ認証結果識別子生成手段 2 7 0 1 を有している形態になっている。

40

【 0 1 1 4 】

アプリ認証結果識別子生成手段 2 7 0 1 は、T R M アクセスライブラリ部 4 0 1 によるアプリケーションの認証の成功を条件としてアプリ認証結果識別子 2 7 0 2 を生成する。ここに、「アプリケーション」とは、ダウンロード部 1 0 2 にダウンロードされたアプリケ

50

ーションである。「T R Mアクセスライブラリ部 4 0 1 によるアプリケーションの認証」とは、アプリケーションに付された署名と、T R Mアクセスライブラリ部 4 0 1 が生成した署名認証用ダイジェストに基づいて行なわれる認証を意味する。

【 0 1 1 5 】

本実施の形態において、認証モジュール内アプリケーション 2 1 0 4 は、認証の成功を示すアプリ認証結果識別子の存在を条件として、端末アプリケーションに対して認証モジュール内アプリケーション 2 1 0 4 に対するアクセスを可能とし、認証モジュール内アプリケーション 2 1 0 4 は端末アプリケーションからのアクセスを受け入れる。

【 0 1 1 6 】

例えば、最初に端末アプリケーション 2 1 0 2 が動作をしており、それに対して、認証モジュール内アプリケーションがまだ動作をしていない場合、認証モジュール内アプリケーションを起動する際には、アプリ認証結果識別子 2 7 0 2 の存在する場合に限り、認証モジュール内アプリケーションが起動される。あるいは、端末アプリケーション 2 1 0 2 と認証モジュール内アプリケーション 2 1 0 4 との両方が起動している場合には、端末アプリケーション 2 1 0 2 から認証モジュール内アプリケーション 2 1 0 4 へアクセスが発生した場合には、アプリ認証結果識別子 2 7 0 2 の存在する場合に限り、そのアクセスを受け入れる。

【 0 1 1 7 】

端末 1 0 0 内で認証モジュール内アプリケーションにアクセスする端末アプリケーションが 1 つだけ動作するのであれば、アプリ認証結果識別子 2 7 0 2 は一種類あれば十分である。しかし、そのような端末アプリケーションが複数個、端末 1 0 0 内で動作するのであれば、どの端末アプリケーションが認証されたかを示すために、端末アプリケーション別にアプリ認証結果識別子がアプリ認証結果識別子生成手段 2 7 0 1 により生成されることになる。あるいは、2 個以上の端末アプリケーションが同時に認証モジュール内アプリケーションにアクセスすることがないと保証される場合には、アプリ認証識別子は一種類だけ生成されることとし、認証された端末アプリケーションが認証モジュール内アプリケーションにアクセスする瞬間だけ、そのアプリ認証識別子が生成され、認証された端末アプリケーションによる認証モジュール内アプリケーションのアクセスが終了すると、そのアプリ認証識別子を削除するようにしてもよい。

【 0 1 1 8 】

このような実施の形態により、認証を受けた端末アプリケーションのみが認証モジュール内アプリケーションにアクセスすることができ、認証モジュール 1 0 1 のセキュリティが確保される。

【 0 1 1 9 】

(実施の形態 1 1)

図 2 8 は、本発明の実施の形態 1 1 にかかわるアプリケーション認証システムの機能ブロック図を示す。本実施の形態におけるアプリケーション認証システムは、端末と、認証モジュールと、アプリケーションを端末にダウンロードするサーバと、からなる。

【 0 1 2 0 】

図 2 8 において、端末 2 8 0 1 は、ダウンロード部 2 8 0 4 を有する。ダウンロード部 2 8 0 4 は、アプリケーションをダウンロードする部である。例えば、サーバ 2 8 0 3 よりアプリケーションをダウンロードする。

【 0 1 2 1 】

認証モジュール 2 8 0 2 は、T R M部 2 8 0 5 を有する。T R M部 2 8 0 5 は、アプリケーションの認証の処理のための情報を耐タンパ領域に保持する。ここで、「アプリケーション」とは、端末 2 8 0 1 のダウンロード部 2 8 0 4 にダウンロードされたアプリケーションを意味する。「アプリケーションの認証」とは、アプリケーションが、信頼のおける者によって発行されたかどうか、不正な動作をしないことの保証を受けているものであるかどうか、あるいは、信頼のおける者から発行されてからの改ざん又は不正な動作をしないことが保証されてからの改ざんがされていないかどうか、などアプリケーションが不正

10

20

30

40

50

な動作をしないことを確認することである。この処理の方法としては、通常、S H A - 1 や M D 5 などの、入力データを処理して得られる結果データが一致するような二つの異なる入力データを見つけることが困難なハッシュ関数を用いて、アプリケーションの実行のためのデータを入力データとして処理して得られる結果データを求め、それを暗号化したもの（いわゆる「署名」）が用いられる。従って「アプリケーションの認証の処理のための情報」とは、この署名そのもの、あるいは、署名を復号してハッシュ値を得るのに必要な復号鍵である。「耐タンパ領域」とは、認証モジュールの記憶領域であって、その記憶領域のデータを不正に読み出すことや、その記憶領域のデータを不正に書き換えることが困難な記憶領域である。例えば、その記憶領域にアクセスするためには、正しい手順を行わないとアクセスできないハードウェアを経由しなければならないようにしたり、記憶領域に記憶されているデータが暗号化されているようにしたりする。

10

【 0 1 2 2 】

サーバ 2 8 0 3 は、端末認証部 2 8 0 6 を有する。端末認証部 2 8 0 6 は、端末 2 8 0 1 を介した認証モジュール 2 8 0 2 の認証が成功することを条件に端末 2 8 0 1 の認証が成功したと判断する。すなわち、サーバ 2 8 0 3 は、認証モジュール 2 8 0 2 を認証することを行なう。その際、サーバ 2 8 0 3 と認証モジュール 2 8 0 2 とは通信を行なう必要があるが、その通信は、端末 2 8 0 1 を中継して行なわれる。サーバ 2 8 0 3 が認証モジュール 2 8 0 2 の認証を行なう方法としては、サーバ 2 8 0 3 は、乱数を発生し、その乱数を認証モジュールの公開鍵によって暗号化を行い、端末 2 8 0 1 を介して認証モジュール 2 8 0 2 へ、暗号化された乱数を復号するように要求する。認証モジュール 2 8 0 2 は、耐タンパ領域に格納されている認証モジュール 2 8 0 2 の秘密鍵を用いた復号によりサーバ 2 8 0 3 が発生した乱数を取得し、それを端末 2 8 0 1 の中継によりサーバ 2 8 0 3 へ返す。サーバ 2 8 0 3 は、発生した乱数と、認証モジュール 2 8 0 2 から送られてきた乱数が等しいかどうかを判断して認証を行なう。あるいは、サーバ 2 8 0 3 は、乱数をそのまま認証モジュール 2 8 0 2 へ送り、認証モジュール 2 8 0 2 は、秘密鍵によって暗号化を行い、その結果をサーバ 2 8 0 3 へ返し、サーバ 2 8 0 3 は、認証モジュール 2 8 0 2 の公開鍵によって復号を行い、認証モジュール 2 8 0 2 へ送った乱数と等しいかどうかで認証モジュールを認証する方法もある。

20

【 0 1 2 3 】

認証モジュール 2 8 0 2 は端末 2 8 0 1 に装着されるものであるので、サーバ 2 8 0 3 が認証モジュール 2 8 0 2 を認証することにより、認証モジュール 2 8 0 2 が装着された端末 2 8 0 1 も認証されることとなる。また、端末 2 8 0 1 の持つ固有な情報、例えば、端末 2 8 0 1 の製造番号や、機器の種別を示す識別子、端末 2 8 0 1 の R O M に格納された識別子、バージョン番号が認証モジュールの耐タンパ領域に存在することを条件として認証モジュール 2 8 0 2 が端末 2 8 0 1 を認証することにすれば、認証がより確かなものとなる。

30

【 0 1 2 4 】

これにより、サーバ 2 8 0 3 は、端末 2 8 0 1 に耐タンパ領域が無くても端末 2 8 0 1 を認証することが可能となり、サーバ 2 8 0 3 は、端末 2 8 0 1 を正しく認証することができ、サーバ 2 8 0 3 と端末 2 8 0 1 の間で課金処理、決済処理などが行なえる。また、サーバ 2 8 0 3 から端末 2 8 0 1 へ機密性の高いデータを含むアプリケーションをダウンロードすることが可能となり、複雑な商取引操作が本実施の形態におけるアプリケーション認証システムにおいて実行することが可能となる。

40

【 0 1 2 5 】

(実施の形態 1 2)

図 2 9 は、本発明の実施の形態 1 2 に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態においては、アプリケーション認証システムは、実施の形態 1 1 のように端末 2 8 0 1 、認証モジュール 2 8 0 2 、アプリケーションを端末 2 8 0 1 にダウンロードするサーバ 2 8 0 3 と、からなる。

【 0 1 2 6 】

50

端末 2801 は、ダウンロード部 2804 と T R M アクセスライブラリ部 2901 とを有している。ダウンロード部 2804 は、アプリケーションをダウンロードする。この場合、アプリケーションは、サーバ 2803 よりダウンロードされる。あるいは、サーバ 2803 以外よりダウンロードされてもよい。T R M アクセスライブラリ部 2901 は、認証モジュール 2802 に自身が認証されることを条件としてアプリケーションの認証のための処理をする。すなわち、T R M アクセスライブラリ部 2901 は、認証モジュール 2802 に自身を認証させ、正しく認証が行なわれると、ダウンロード部 2804 にダウンロードされたアプリケーションの認証のための処理を行なう。T R M アクセスライブラリ部 2901 が、認証モジュール 2802 に自身を認証させる方法としては、端末 2801 の固有な情報、例えば、製造番号や、種類を示す識別子、あるいは、T R M アクセスライブラリ部 2901 を実現するソフトウェアのシリアル番号、バージョン番号などを認証モジュール 2802 に出力し、認証モジュール 2802 の耐タンパ領域に出力された端末 2801 の固有な情報やシリアル番号、バージョン番号などが存在するかどうかで行なう方法がある。

10

【0127】

認証モジュール 2802 は、T R M 部 2805 と、T R M アクセスライブラリ部認証部 2902 とを有する。T R M 部 2805 は、T R M アクセスライブラリ部 2901 を認証するための情報である T R M アクセスライブラリ部認証情報を耐タンパ領域に保持する。「T R M アクセスライブラリ部認証情報」としては上述のように、端末 2801 の固有な情報、例えば、製造番号や、種類を示す識別子、あるいは、T R M アクセスライブラリ部 2901 を実現するソフトウェアのシリアル番号、バージョン番号などが挙げられる。T R M アクセスライブラリ部認証部 2902 は、T R M アクセスライブラリ部認証情報に基づいて端末 2801 の T R M アクセスライブラリ部 2901 を認証する。この認証の方法としては、上述のように、T R M アクセスライブラリ部 2901 より出力された端末 2801 の固有な情報や T R M アクセスライブラリ部のシリアル番号、バージョン番号などの識別情報を T R M アクセスライブラリ部認証部 2902 が受け取り、T R M 部 2805 にその識別情報が存在するかどうかで行なう方法がある。この認証の結果は、T R M アクセスライブラリ部認証部 2902 より T R M アクセスライブラリ部 2901 へ出力される。また、この認証の結果は、認証モジュール 2802 内に保持され、その後の端末との情報の交換の際に参照され、T R M アクセスライブラリ部 2901 が正しく認証されていれば、認証モジュール 2802 は正しい情報の交換を行い、そうでなければ、正しくない情報の交換を行なうようにする。

20

30

【0128】

サーバ 2803 は、サーバ T R M アクセスライブラリ部認証部 2903 を有する。サーバ T R M アクセスライブラリ部認証部 2903 は、端末 2801 のアクセスライブラリ部 2901 を介する認証モジュール 2802 の T R M 部 2805 の認証が成功することを条件として T R M アクセスライブラリ部 2901 の認証が成功したと判断する。サーバ T R M アクセスライブラリ部認証部 2903 が、端末 2801 のアクセスライブラリ部 2901 を介する認証モジュール 2802 の T R M 部 2805 の認証を行なう方法としては、次のものがある。すなわち、サーバ 2803 は、乱数を発生し、その乱数を認証モジュール 2802 の公開鍵によって暗号化を行い、端末 2801 を介して認証モジュール 2802 へ、暗号化された乱数を復号するように要求する。認証モジュール 2802 は、耐タンパ領域に格納されている自身の秘密鍵を用いて、乱数を復号し、それを端末 2801 を介してサーバ 2803 へ返す。サーバ 2803 は、発生した乱数と、認証モジュール 2802 から送られてきた乱数が等しいかどうかを判断して認証を行なう。あるいは、サーバ 2803 は、乱数をそのまま認証モジュール 2802 へ送り、認証モジュール 2802 は、自身の秘密鍵によって暗号化を行い、その結果をサーバ 2803 へ返し、サーバ 2803 は、認証モジュール 2802 の公開鍵によって復号を行い、認証モジュール 2802 へ送った乱数と等しいかどうかで認証モジュールを認証する方法もある。

40

【0129】

50

図30は、サーバTRMアクセスライブラリ部認証部2903と、TRMアクセスライブラリ部2901と、認証モジュール2802との相互作用を説明するシーケンス図である。ステップS3001において、TRMアクセスライブラリ部2901より、自身を認証する要求が認証モジュール2802へ出力され、ステップS3002において、認証モジュールでの認証結果が出力される。ステップS3003において、サーバTRMアクセスライブラリ部認証部2903よりTRMアクセスライブラリ部2901へ認証要求が出力され、これに対応して、ステップS3004において、認証要求がTRMアクセスライブラリ部2901より認証モジュール2802へ出力され、ステップS3005において、認証モジュールは、自身がサーバTRMアクセスライブラリ部認証部2903へ認証されるように結果を返す。この際、TRMアクセスライブラリ部2901が正しく認証されているかどうかによって、正しい結果を返したり、正しくない結果を返したりする。ステップS3006において、TRMアクセスライブラリ部2901は、サーバTRMアクセスライブラリ部認証部2903へ、認証モジュール2802から出力された結果を返す。サーバTRMアクセスライブラリ部認証部2903は、この結果を調べ、認証モジュール2802が認証できれば、TRMアクセスライブラリ部2901も認証されたものと判断することになる。

10

【0130】

上述のように、TRMアクセスライブラリ部2901のTRMアクセスライブラリ部認証部2902による認証の結果は認証モジュール2802内に保持され、その認証の結果に応じて、認証モジュール2802は正しい情報の交換を行ったり、行なわなかったりするので、サーバ2803がTRMアクセスライブラリ部2901を介して認証モジュールのTRM部の認証を行い、正しく認証ができれば、TRMアクセスライブラリ部2901の認証が成功したと判断してよいことになる。

20

【0131】

これにより、サーバ2803は、端末2801に耐タンパ領域が無くても端末2801を認証することが可能となり、サーバ2803は、端末2801を正しく認証することができ、サーバ2803と端末2801の間で課金処理、決済処理などが行なえる。また、サーバ2803から端末2801へ機密性の高いデータを含むアプリケーションをダウンロードすることが可能となり、複雑な商取引操作が本実施の形態におけるアプリケーション認証システムにおいて実行することが可能となる。

30

【0132】

また、TRMアクセスライブラリ部2901がTRMアクセスライブラリ部認証部2902により認証されると、TRMアクセスライブラリ部2901は、認証モジュールにより信頼できるものと判断される。これにより、ダウンロード部2804にダウンロードされたアプリケーションの認証の処理の全部または一部をTRMアクセスライブラリ部2901に行なうようにすると、TRMアクセスライブラリ部2901によるアプリケーションの認証の処理の全部または一部の結果は認証モジュール2802にとって信頼できるものとなる。したがって、TRMアクセスライブラリ部2901によるアプリケーションの認証の処理の全部または一部の結果を用いて、認証モジュール2802はダウンロード部2804にダウンロードされたアプリケーションの認証を行なうことが可能となる。この結果、正しく認証されたアプリケーションに、認証モジュール内のデータのアクセスを許可することができ、複雑な商取引の操作を行なうことができるようになる。

40

【0133】

なお、TRMアクセスライブラリ部2901によるアプリケーションの認証のための処理は、アプリケーションが認証モジュール2802のTRM部2805の耐タンパ領域へアクセスしたことを条件に行なうようにしてもよい。これにより、耐タンパ領域にアクセスしないアプリケーションの認証を行なう必要がなくなる。

【0134】

また、TRMアクセスライブラリ部2901は、アプリケーションの認証のための処理を、アプリケーションがダウンロード部2804にダウンロードされたことを条件に行なう

50

ようにしてもよい。これにより、ダウンロードされたアプリケーションが全て認証され、不正なアプリケーションが端末 2801 で実行される虞がなくなる。

【0135】

また、T R M アクセスライブラリ部 2901 は、アプリケーション認証のための処理を、アプリケーションの実行の開始をトリガとして行なうようにしてもよい。これにより、ダウンロードされたが、実行されないアプリケーションの認証を省略することができる。

【0136】

(実施の形態 13)

図 31 は、本発明の実施の形態 13 に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態に関するアプリケーション認証システムは、端末 2801 と、
10 認証モジュール 2802 と、アプリケーションを端末 2801 にダウンロードするサーバ 2803 とからなる。

【0137】

端末 2801 は、ダウンロード部 2804 と、T R M アクセスライブラリ部 2901 と、を有し、T R M アクセスライブラリ部 2901 は、署名用ダイジェスト生成手段 3101 と、ダウンロードアプリケーション署名取得手段 3102 と、アプリ認証データ出力手段 3103 とを備えている。

【0138】

ダウンロード部 2804 は、アプリケーション 3104 をダウンロードする。アプリケーション 3104 は、サーバ 2803 よりダウンロードされてもよい。また、サーバ 2803 以外、例えば、認証モジュール 2802 からダウンロードされてもよい。本実施の形態
20 においてはアプリケーション 3104 は、アプリケーション 3104 の署名 3105 と共にダウンロードされるものとする。「署名 3105 と共にダウンロードされる」とは、同時にダウンロードされるということのみならず、アプリケーション 3104 のダウンロードと署名 3105 のダウンロードとは前後して行なわれてもよく、後述のアプリケーション 3104 の認証が行なわれる時までには、アプリケーション 3104 と署名 3105 とがダウンロードされていることを意味する。

【0139】

署名用ダイジェスト生成手段 3101 は、アプリケーションから署名用ダイジェストを生成する。すなわち、ダウンロード部 2804 にダウンロードされたアプリケーション 3104 から署名用ダイジェストを生成する。「署名用ダイジェスト」とは、署名 3105 を生成するときに使われたハッシュ関数と同じハッシュ関数を用いて得られる値である。
30

【0140】

ダウンロードアプリケーション署名取得手段 3102 は、アプリケーション 3104 のダウンロードと共にダウンロードされた署名 3105 を取得する。上述のように「アプリケーション 3104 のダウンロードと共にダウンロードされた」とは、同時にダウンロードされたということのみならず、後述のアプリケーション 3104 の認証が行なわれる時までには、アプリケーション 3104 と署名 3105 とのダウンロードが終了していることを意味する。

【0141】

アプリ認証データ出力手段は、取得した署名と、署名用ダイジェスト生成手段 3101 とによって生成された署名用ダイジェストと、をサーバに送信する。「取得した署名」とは、ダウンロードアプリケーション署名取得手段 3102 により取得された署名 3105 である。
40

【0142】

署名用ダイジェスト生成手段 3101 と、ダウンロードアプリケーション署名取得手段 3102 と、アプリ認証データ出力手段 3103 とは、ダウンロード部 2804 にダウンロードされたアプリケーション 3104 を認証するための処理を行なう。この処理は、T R M アクセスライブラリ部 2901 が認証モジュール 2802 により認証されたことを条件として行なわれるようにしてもよい。
50

【 0 1 4 3 】

認証モジュールは、T R M部 2 8 0 5とT R Mアクセスライブラリ部認証部 2 9 0 2と、を有している。T R M部 2 8 0 5とT R Mアクセスライブラリ部認証部 2 9 0 2とは、実施の形態 1 2のT R M部とT R Mアクセスライブラリ部と同じものである。

【 0 1 4 4 】

サーバ 2 8 0 3は、サーバT R Mアクセスライブラリ部認証部 2 9 0 3と、アプリ認証データ入力部 3 1 0 6と、サーバアプリ認証部 3 1 0 7とを有する。

【 0 1 4 5 】

サーバT R Mアクセスライブラリ部認証部 2 9 0 3は、実施の形態 1 2におけるものと同じであり、端末 2 8 0 1のT R Mアクセスライブラリ部 2 9 0 1を介する認証モジュールのT R M部 2 8 0 5の認証が成功することを条件としてT R Mアクセスライブラリ部 2 9 0 1の認証が成功したと判断する。

10

【 0 1 4 6 】

アプリ認証データ入力部 3 1 0 6は、サーバT R Mアクセスライブラリ部認証部 2 9 0 3により認証が成功したと判断されたT R Mアクセスライブラリ部 2 9 0 1のアプリ認証データ出力手段 3 1 0 3から出力された署名用ダイジェストと、署名と、を入力する。

【 0 1 4 7 】

サーバアプリ認証部 3 1 0 7は、アプリ認証データ入力部 3 1 0 6に入力された署名用ダイジェストと、署名と、に基づいてアプリケーションの認証を行なう。認証は、署名を復号してダイジェストを求め、そのダイジェストが署名用ダイジェストと等しいかどうかを判断することにより行なわれる。もし、署名が公開鍵暗号化方式のサーバ 2 8 0 3の秘密鍵で暗号化されている場合には、アプリ認証データ入力部 3 1 0 6に入力された署名用ダイジェストをサーバ 2 8 0 3の秘密鍵で暗号化し、得られたものがアプリ認証データ入力部 3 1 0 6に入力された署名と等しいかどうかで判断するようにしてもよい。

20

【 0 1 4 8 】

図 3 2は、本実施の形態におけるアプリケーション認証システムを構成するサーバ 2 8 0 3、端末 2 8 0 1、認証モジュール 2 8 0 2の時間に沿った相互作用を示す。ステップ S 3 2 0 1からステップ S 3 2 0 6までは、実施の形態 1 2における図 3 0におけるステップ S 3 0 0 1からステップ S 3 0 0 6までと同じである。ステップ S 3 2 0 6の後、アプリケーション 3 1 0 4がダウンロード部 2 8 0 4にダウンロードされると、署名用ダイジェスト生成手段 3 1 0 1により、アプリケーション 3 1 0 4の署名用ダイジェストが生成され、ダウンロードアプリケーション署名取得手段 3 1 0 2により、署名 3 1 0 5が取得され、アプリ認証データ出力手段 3 1 0 3により、アプリ認証データ入力部 3 1 0 6へ署名用ダイジェストと署名が入力される（ステップ S 3 2 0 7）。その後、サーバアプリ認証部 3 1 0 7によりアプリ認証データ入力部 3 1 0 6へ入力された署名用ダイジェストと署名によりアプリケーション 3 1 0 4の認証が行なわれる。

30

【 0 1 4 9 】

本実施の形態によれば、サーバ 2 8 0 3のサーバT R Mアクセスライブラリ部認証部 2 9 0 3が、端末 2 8 0 1のT R Mアクセスライブラリ部 2 9 0 1を介する認証モジュール 2 8 0 2のT R M部 2 8 0 5の認証が成功することを条件として端末 2 8 0 1のT R Mアクセスライブラリ部 2 9 0 1の認証が成功したと判断する。このため、サーバ 2 8 0 3は、T R Mアクセスライブラリ部 2 9 0 1のアプリ認証データ出力手段 3 1 0 3がアプリ認証データ入力部 3 1 0 6へ送信するアプリケーション 3 1 0 4の署名用ダイジェストと署名 3 1 0 5が、実際にアプリケーション 3 1 0 4に由来するものであると判断することができ、サーバは、アプリケーション 3 1 0 4を認証することができる。

40

【 0 1 5 0 】

（実施の形態 1 4）

図 3 3は、本発明の実施の形態 1 4に関するアプリケーション認証システムの機能ブロック図を示す。本実施の形態 1 3においては、ダウンロード部 2 8 0 4にダウンロードされたアプリケーションの認証の処理の一部がサーバ 2 8 0 3で実行されるようになっていた

50

が、本実施の形態においては、サーバ２８０３以外でアプリケーションの認証が行なわれ、サーバ２８０３は認証の結果だけを取得するようになっている。

【０１５１】

端末２８０１は、ダウンロード部２８０４とＴＲＭアクセスライブラリ部２９０１とを有している。ダウンロード部２８０４は、アプリケーションをダウンロードする。ＴＲＭアクセスライブラリ部２９０１は、認証成功情報生成手段３３０１と認証成功情報出力手段３３０３とを備えている。

【０１５２】

認証成功情報生成手段３３０１は、アプリケーションの認証の成功を示す認証成功情報３３０２を生成する。本実施の形態において、アプリケーションの認証は、ＴＲＭアクセスライブラリ部２９０１内のみで行なわれるようになっていてもよい。また、ＴＲＭアクセスライブラリ部２９０１と認証モジュール２８０２とが協働して行なうようになっていてもよく、認証成功情報生成手段３３０１は、その認証の結果を取得し、認証が成功したかどうかを示す認証成功情報３３０２を生成する。このとき、認証成功情報３３０２を認証モジュール２８０２の秘密鍵やサーバ２８０３の公開鍵により認証成功情報３３０２が暗号化されるようにしてもよい。

10

【０１５３】

認証成功情報出力手段３３０３は、認証成功情報生成手段３３０１にて生成された認証成功情報３３０２を出力する。もし、認証成功情報３３０２が暗号化されていない場合には、認証成功情報３３０２を認証モジュール２８０２の秘密鍵やサーバ２８０３の公開鍵により認証成功情報３３０２を暗号化して出力するようになしてもよい。

20

【０１５４】

認証モジュール２８０２は、ＴＲＭ部２８０５と、ＴＲＭアクセスライブラリ部認証部２９０２と、を有し、実施の形態１３と同じ動作をする。

【０１５５】

サーバ２８０３は、サーバＴＲＭアクセスライブラリ部認証部と、認証成功情報入力部３３０４と、サーバアプリ認証部３３０５とを有している。

【０１５６】

サーバＴＲＭアクセスライブラリ部認証部は、実施の形態１３と同じ動作をする。

【０１５７】

30

認証成功情報入力部３３０４は、サーバＴＲＭアクセスライブラリ部認証部により認証が成功したと判断されたＴＲＭアクセスライブラリ部の認証成功情報出力手段から出力された認証成功情報を入力する。ＴＲＭアクセスライブラリ部２９０１がサーバＴＲＭアクセスライブラリ部認証部により認証が成功したと判断された場合には、ＴＲＭアクセスライブラリ部２９０１と認証モジュール２８０２とにより出力される情報は、サーバ２８０３にとって信頼のおけるものであるので、認証成功情報出力手段３３０３の内容は、信頼のおけるものと判断することができる。

【０１５８】

サーバアプリ認証部３３０５は、認証成功情報入力部３３０４に入力された認証成功情報に基づいてアプリケーションの認証を行なう。例えば、認証成功情報出力手段３３０３により出力された認証成功情報が、認証モジュール２８０２の秘密鍵やサーバ２８０３の公開鍵で暗号化されている場合には、復号を行い、認証成功情報の内容を調べてダウンロード部２８０４にダウンロードされたアプリケーションを認証する。

40

【０１５９】

（実施の形態１５）

本発明においては、アプリケーション（アプリケーションプログラム）を認証するために、アプリケーションのダウンロードを行なうと共に、署名をダウンロードすることが必要となる。以下では、アプリケーションの中に、アプリケーションの署名を格納したアプリケーションを説明する。

【０１６０】

50

アプリケーションプログラムは、通常、アプリケーション本体と、アプリケーション定義ファイルに分けることができる。「アプリケーション本体」とは、アプリケーションプログラムを実行するためのコードやデータであり、「アプリケーション定義ファイル」とは、アプリケーション本体の属性を記述するファイルであり、「アプリケーション本体の属性」としては、例えば、アプリケーション本体の大きさ、アプリケーションプログラムを実行するためのエントリポイント、アプリケーションプログラムの実行時にアプリケーションプログラムに渡されるべきパラメータ（Java（登録商標）においては、メインクラスの起動パラメータ）などがある。アプリケーション定義ファイルにおいて、アプリケーション本体の属性を記述する部分を「属性情報格納部」と呼ぶことにすると、属性情報格納部に、アプリケーションの作成者が自由に利用できるオブショナル領域が存在する場合がある。そこで、このオブショナル領域に、アプリケーション本体の署名データを格納するようにしてもよい。なお、アプリケーション本体は、コードとデータそのものではなく、コードとデータとを圧縮したものであってもよい。同様に、アプリケーション定義ファイルもアプリケーション本体の属性の記述を圧縮したものであってもよい。

10

【0161】

図34は、Java（登録商標）アプリケーション、特に、iアプリのアプリケーションの構造を例示している。iアプリにおいては、アプリケーション本体はJARファイル3401に格納され、アプリケーション定義ファイルは、ADFファイル3402として提供されるようになっている。ADFファイル3402に格納されたアプリケーション本体の属性は、アプリケーションの名前としてApp Nameという必須キーにより示され、アプリケーション本体のサイズはApp Sizeという必須キーにより示される。さらに、アプリケーションの作成者が自由に利用できるオブショナル領域として、App Paramというオブショナルキーで示されるものがある。このApp Paramで示される領域は、最長255バイト利用可能である。一方、アプリケーション本体の署名は、160ビットの楕円暗号を用いると20バイト必要であり、1024ビットのRSA暗号を用いるのであれば、128バイト必要であり、App Paramで示される領域に収まることになる。よってアプリケーション本体の署名をApp Paramで示される領域に格納することが可能となる。

20

【0162】

図35は、このように、オブショナル領域にアプリケーション本体の署名データを格納したアプリケーションを認証する場合の動作を説明するフローチャートである。ステップS3501において、オブショナル領域から署名データを取得する。ステップS3502において、ステップS3501で取得された署名データを利用して、署名を検証することを行なう。これらのステップは、プログラムによって実行可能である。

30

【0163】

また、図34は、アプリケーションプログラムのデータ構造を示しているとみなすこともできる。すなわち、コード及びデータの圧縮ファイルであるJARファイルを格納するJARファイル部3401とアプリケーションの定義ファイルであるADFファイルを格納するADFファイル部3402とからなるデータ構造とみなすこともできる。このようなデータ構造において、ADFファイル部3402のADFファイルには、メインクラスの起動パラメータなどを格納するApp Paramで示される領域があり、App Paramで示される領域に、JARファイル部3401に格納されているJARファイルの署名が格納されている。

40

【0164】

App Paramで示される領域に格納されるJARファイルの署名は、アプリケーションの動作を保証する者による署名であってもよい。ここに、「アプリケーションの動作を保証する者」とは、JARファイルに格納されたコードとデータにより動作するアプリケーションを作成した者、そのアプリケーションを配布する者、そのアプリケーションの作成した者、そのアプリケーションの動作を検証し不正な動作をしないことを保証する第三者などである。

50

【 0 1 6 5 】

図 3 4 に示したアプリケーションプログラムのデータ構造は、ビット列（ビットストリーム）によって表現可能であるので、このようなビット列を記録した（ＳＤ）メモリカードや、フロッピー（登録商標）ディスク、コンパクトディスクなどの記録媒体を作成することができる。

【 0 1 6 6 】

このようなアプリケーションプログラムにより、アプリケーションをダウンロードするとアプリケーション本体のみならず、アプリケーション本体の署名もダウンロードされることになり、別に署名をダウンロードする手間を省くことができる。

【 0 1 6 7 】

10

（実施の形態 1 6）

図 3 7 は、本発明の実施の形態 1 6 に関する端末の機能ブロック図を例示する。本実施の形態の特徴は、実施の形態 1 のアプリケーション認証システムにおける端末の内部に認証モジュールを備えるようにし、一体化したことである。

【 0 1 6 8 】

本実施の形態において、端末 3 7 0 0 は、ダウンロード部 3 7 0 1 と、ＴＲＭ部 3 7 0 2 とを有する。

【 0 1 6 9 】

ダウンロード部 3 7 0 1 は、アプリケーションをダウンロードする。すなわち、実施の形態 1 におけるダウンロード部 1 0 2 と同じ機能を有する。

20

【 0 1 7 0 】

ＴＲＭ部 3 7 0 2 は、アプリケーションの認証のための情報を耐タンパ領域に保持する。すなわち、実施の形態 1 における認証モジュール 1 0 1 内のＴＲＭ部 1 0 3 と同じ機能を有する。

【 0 1 7 1 】

したがって、本実施の形態の端末においては、アプリケーションのダウンロードの手順や、ダウンロードされたアプリケーションの認証の手順は、実施の形態 1 におけるものと同じであってよい。

【 0 1 7 2 】

このような端末を用いることにより、例えば、サービス提供会社から端末 3 7 0 0 のダウンロード部 3 7 0 1 にダウンロードされたアプリケーションに対して認証の処理を行い、その認証の処理が成功した場合に、アプリケーションに対して、ＴＲＭ部 3 7 0 2 の保持された情報などの端末に格納された情報へのアクセスを安全に許可することが可能となる。

30

【 0 1 7 3 】

なお、「端末」という語を用いたが、これは、携帯電話に代表される携帯可能な端末などに限定されることを意味しない。例えば、家庭用電化製品であってもよいし、いわゆる、情報家電やネット家電と呼ばれるものであってもよい。そのような製品を例示列挙すれば、エアコンディショナ、加湿器、除湿器、空気清浄機、電子レンジ、オーブン、冷蔵庫、食器洗い機、湯沸し器、アイロン、ズボンプレスサー、電気掃除機、洗濯機、乾燥機、電気毛布、電気敷布、照明機器、テレビ受像機、ラジオ受信機、テープレコーダなどのオーディオ機器、カメラ、ＩＣレコーダ、電話機、ファクシミリ送受信機、コピー機、プリンター、スキャナー、パーソナルコンピュータ、などを挙げることができる（このことは、次に説明する実施の形態 1 7 における、「端末」についても言える）。

40

【 0 1 7 4 】

（実施の形態 1 7）

図 3 8 は、本発明の実施の形態 1 7 に関する端末の機能ブロック図を例示する。本実施の形態の特徴は、実施の形態 2 などのアプリケーション認証システムにおける端末の内部に認証モジュールを備えるようにして、端末と認証モジュールとを一体化したことである。

【 0 1 7 5 】

50

本実施の形態において、端末 3800 は、認証モジュール 3801 を備えた端末となっており、ダウンロード部 3802 と、T R M アクセスライブラリ部 3803 と、を有している。認証モジュール 3801 は、T R M 部 3804 と、T R M アクセスライブラリ部認証部 3805 と、を有している。

【0176】

認証モジュール 3801 は、耐タンパ領域に情報を保持しその情報を用いて認証のための処理を行なう。詳細については、後述する。

【0177】

ダウンロード部 3802 は、アプリケーションをダウンロードする。すなわち、実施の形態 2 などにおけるダウンロード部 102 と同じ機能を有する。

10

【0178】

T R M アクセスライブラリ部 3803 は、認証モジュール 3801 に自身が認証されることを条件としてアプリケーションの認証のための処理をする。すなわち、実施の形態 2 などにおける T R M アクセスライブラリ部 401 と同じ機能を有する。なお、「認証モジュール 3801 に自身が認証される」とは、後述のように、T R M アクセスライブラリ部認証部 3805 により認証がされることを意味する。

【0179】

T R M 部 3804 は、T R M アクセスライブラリ部認証情報を前記耐タンパ領域に保持する。「T R M アクセスライブラリ認証情報」とは、T R M アクセスライブラリ部を認証するための情報であり、実施の形態 2 の定義と同じである。したがって、T R M 部 3804 は、実施の形態 2 などの T R M 部 103 と同じ機能を有する。なお、耐タンパ領域は、T R M 部 3804 の内部にあってもよいし、認証モジュール 3801 の内部であって T R M 部 3804 の外部にあってもよい。

20

【0180】

T R M アクセスライブラリ部 3805 は、T R M アクセスライブラリ部認証情報に基づいて T R M アクセスライブラリ部 3803 を認証する。したがって、T R M アクセスライブラリ部 3805 は、実施の形態 2 などの T R M アクセスライブラリ部 402 と同じ機能を有する。

【0181】

図 39 は、本実施の形態における端末 3800 の処理の流れを説明するフローチャートを例示する。このフローチャートに例示された処理においては、ダウンロード部 3802 にアプリケーションがダウンロードされていることが仮定されている。

30

【0182】

ステップ S 3901 において、T R M アクセスライブラリ部認証部 3805 により、T R M アクセスライブラリ部 3803 の認証の処理を行なう。

【0183】

ステップ S 3902 において、T R M アクセスライブラリ部 3803 が認証されたかが判断され、もし認証されたのであれば、ステップ S 3903 へ処理が進められる。図 39 では、もし認証されなければ、処理が終了することになっているが、その代わりに、ダウンロード部 3802 にダウンロードされたアプリケーションの破棄が行なわれるようになっていてもよい。

40

【0184】

ステップ S 3903 において、ダウンロード部 3802 にダウンロードされたアプリケーションの認証の処理を、T R M アクセスライブラリ部 3803 により行なう。

【0185】

ステップ S 3904 において、ダウンロードされたアプリケーションが認証されたのであれば、ステップ S 3905 へ処理が進められる。もし、認証されなければ、処理が終了する。処理が終了する代わりに、ダウンロード部 3802 にダウンロードされたアプリケーションの破棄が行なわれるようになっていてもよい。

【0186】

50

ステップ S 3 9 0 5 において、ダウンロードされたアプリケーションによる認証モジュールへのアクセスが許可される。この許可を行なうのは、認証モジュール 3 8 0 1 である。あるいは、アプリケーションが認証モジュール 3 8 0 1 へアクセスする場合に、必ず、T R M アクセスライブラリ部 3 8 0 3 の機能を利用するのであれば、T R M アクセスライブラリ部 3 8 0 3 により認証モジュールへのアクセスが許可されるようになっていてもよい。

【 0 1 8 7 】

なお、実施の形態 2 において述べたように、T R M アクセスライブラリ部 3 8 0 2 は、アプリケーションマネージャ、デバイスドライバなどを備えていてもよい。

【 0 1 8 8 】

本実施の形態によれば、端末の内部に認証モジュールを備えることにより、端末の内部に高度に保護されるべき情報を保持することが可能となり、端末にダウンロードされたアプリケーションの認証を行なうことが可能となる。

【 0 1 8 9 】

(実施の形態 1 8)

本発明の実施の形態 1 8 は、第一の機器と、認証モジュールと、からなるアプリケーション認証システムに関する。本実施の形態においては、認証モジュールが保持する情報を用いて、第一の機器に格納されたアプリケーションが認証される。

【 0 1 9 0 】

図 4 0 は、本実施の形態のアプリケーション認証システムの機能ブロック図を例示している。アプリケーション認証システムは、第一の機器 4 0 0 1 と、認証モジュール 4 0 0 2 と、からなる。第一の機器は、端末に限られることはなく、例えば、パーソナルコンピュータ、ワークステーション、大型計算機、あるいは、サーバ装置であってもよい。また、第一の機器 4 0 0 1 と認証モジュール 4 0 0 2 とは、電氣的に直接接続されている必要はなく、また、物理的に接触している必要もない。例えば、無線によって接続されていてもよい。また、インターネットに代表されるネットワークにより接続されていてもよい。特に、そのネットワークは、光ケーブルなどの電気の伝導を使用しない媒体を用いて構築されていてもよい。

【 0 1 9 1 】

第一の機器 4 0 0 1 は、アプリケーション格納部 4 0 0 3 を有している。アプリケーション格納部 4 0 0 3 は、アプリケーションを格納する。アプリケーションとは、第一の機器 4 0 0 1 で動作するプログラムに限られない。第一の機器 4 0 0 1 以外の機器で動作するアプリケーションであってもよい。また、「格納する」とは、アプリケーションを保持することである。保持する時間の長さは問わない。また、保持の目的も問わない。例えば、アプリケーションを第一の機器 4 0 0 1 で動作させるための格納が目的であってもよい。また、アプリケーションを第一の機器 4 0 0 1 にダウンロードし動作させることが目的であってもよい。あるいは、アプリケーションが通信で送られる際に、第一の機器 4 0 0 1 が中継するための一時的な保持が目的であってもよい。あるいは、アプリケーションを第一の機器 4 0 0 1 以外の機器にダウンロードするための保持が目的であってもよい。

【 0 1 9 2 】

認証モジュール 4 0 0 2 は、T R M 部 4 0 0 4 を有する。T R M 部 4 0 0 4 は、アプリケーションの認証の処理のための情報を耐タンパ領域に保持する。アプリケーションの認証の処理のための情報としては、例えば、暗号のための秘密鍵やアプリケーションの署名の真正性を確認するための証明書などを挙げることができる。ここでいうアプリケーションとは、第一の機器 4 0 0 1 のアプリケーション格納部 4 0 0 3 に格納されるアプリケーションである。したがって、本実施の形態における認証モジュールの T R M 部 4 0 0 4 は、実施の形態 1 の T R M 部と同じ機能を有するものであってもよい。その場合、アプリケーションの認証の処理の方法などは、実施の形態と同様のものとなる。

【 0 1 9 3 】

本実施の形態のアプリケーション認証システムにより、第一の機器 4 0 0 1 のアプリケー

10

20

30

40

50

ション格納部 4 0 0 3 に格納されているアプリケーションを認証することが可能となるので、例えば、そのアプリケーションが不正な動作をすることを防止することができる。

【 0 1 9 4 】

なお、第一の機器 4 0 0 1 は、家庭用電化製品であってもよい。また、いわゆる、情報家電やネット家電と呼ばれるものであってもよい。そのような製品を例示列举すれば、エアコンディショナ、加湿器、除湿器、空気清浄機、電子レンジ、オーブン、冷蔵庫、食器洗い機、湯沸し器、アイロン、ズボンプレスサー、電気掃除機、洗濯機、乾燥機、電気毛布、電気敷布、照明機器、テレビ受像機、ラジオ受信機、テープレコーダなどのオーディオ機器、カメラ、ＩＣレコーダ、電話機、ファクシミリ送受信機、コピー機、プリンター、スキャナー、パーソナルコンピュータ、などを挙げることができる。

10

【 0 1 9 5 】

(実施の形態 1 9)

本発明の実施の形態 1 9 も、実施の形態 1 8 と同じように、第一の機器と認証モジュールとからなるアプリケーション認証システムに関する。本実施の形態においては、第一の機器に格納されるアプリケーションの認証が行なわれるときには、第一の機器内でその認証を行なう部分が認証モジュールにより認証がされる。

【 0 1 9 6 】

図 4 1 は、本実施の形態におけるアプリケーション認証システムの機能ブロック図を例示している。アプリケーション認証システムは、第一の機器 4 1 0 1 と、認証モジュール 4 1 0 2 と、からなる。なお、実施の形態 1 8 におけるように、第一の機器 4 1 0 1 と認証モジュール 4 1 0 2 とは、電氣的に直接接続されている必要もなく、また、物理的に接触している必要もない。

20

【 0 1 9 7 】

第一の機器 4 1 0 1 は、アプリケーション格納部 4 1 0 3 と、ＴＲＭアクセスライブラリ部 4 1 0 4 と、を有している。

【 0 1 9 8 】

アプリケーション格納部 4 1 0 3 は、アプリケーションを格納する。例えば、実施の形態 1 8 におけるアプリケーション格納部 4 0 0 3 と同じ機能を有している。

【 0 1 9 9 】

ＴＲＭアクセスライブラリ部 4 1 0 4 は、認証モジュール 4 1 0 2 に自身が認証されることを条件としてアプリケーションの認証のための処理をする。ここに「自身」とは、ＴＲＭアクセスライブラリ部 4 1 0 4 を意味する。あるいは、「自身」とは、ＴＲＭアクセスライブラリ部 4 1 0 4 を含む部分であってもよい。例えば、第一の機器 4 1 0 1 自体が「自身」であるとしてもよい。

30

【 0 2 0 0 】

また、「アプリケーション」とは、アプリケーション格納部 4 0 0 3 に格納されたアプリケーションを意味する。

【 0 2 0 1 】

したがって、ＴＲＭアクセスライブラリ部 4 1 0 4 は、自身を認証モジュールによって認証される処理が行なわれ、その処理の結果、正しく認証された場合に、アプリケーション格納部 4 1 0 3 に格納されたアプリケーションの認証の処理を行なう。

40

【 0 2 0 2 】

なお、ＴＲＭアクセスライブラリ部 4 1 0 4 は、実施の形態 2 において説明したように、アプリケーションマネージャとデバイスドライバを備え、それらの処理を行なうようになっていてもよい。

【 0 2 0 3 】

認証モジュール 4 1 0 2 は、ＴＲＭ部 4 1 0 5 と、ＴＲＭアクセスライブラリ部認証部 4 1 0 6 と、を有している。

【 0 2 0 4 】

ＴＲＭ部 4 1 0 5 は、ＴＲＭアクセスライブラリ部認証情報を耐タンパ領域に保持する。

50

「T R Mアクセスライブラリ部認証情報」とは、T R Mアクセスライブラリ部を認証するための情報である。例えば、第一の機器を特定するための情報を挙げることができる。このような情報としては、第一の機器の製造者番号と製造番号、第一の機器に電話番号が割り振られている場合には、その電話番号を例としてあげることができる。また、第一の機器が接続されている他の機器の情報や第一の機器に装備されている部品を特定する情報、あるいは、第一の機器にインストールされているプログラムのバージョン番号など、第一の機器の置かれている状況を示すものであってもよい。また、第一の機器が、なんらかの秘密の情報、例えば、暗号鍵など、を保持できる場合には、その暗号鍵が真正なことを検出するための情報をT R Mアクセスライブラリ部認証情報としてもよい。

【0205】

10

また、耐タンパ領域については、実施の形態2の説明を参照されたい。

【0206】

T R Mアクセスライブラリ部認証部4106は、T R Mアクセスライブラリ部認証情報に基づいて第一の機器4101のT R Mアクセスライブラリ部4104を認証する。すなわち、T R Mアクセスライブラリ部4104から送られてくる情報を取得し、T R Mアクセスライブラリ部4104から得られる情報が、T R Mアクセスライブラリ部認証情報に適合するものであるかどうかを判断し、認証の処理を行なう。

【0207】

なお、T R Mアクセスライブラリ部認証部4106が認証する対象は、T R Mアクセスライブラリ部4104にのみ限られるものではなく、T R Mアクセスライブラリ部4104を含む部分であってもよく、例えば、第一の機器4101の全体を認証するようになっていてもよい。この場合には、T R Mアクセスライブラリ部認証情報は、第一の機器4101の製造番号などや第一の機器4101が置かれている状況などを用いるようになっていてもよい。

20

【0208】

図42は、本実施の形態におけるアプリケーション認証システムの動作を説明するフローチャートを例示する。

【0209】

ステップS4201において、T R Mアクセスライブラリ部4104の認証の処理を、T R Mアクセスライブラリ部認証部4106により行なう。この際、耐タンパ領域に保持されたT R Mアクセスライブラリ部認証情報が用いられる。

30

【0210】

ステップS4202において、ステップS4201の処理によりT R Mアクセスライブラリ部4104が認証されたかどうかを判断する。もし認証されたと判断された場合には、ステップS4203において、アプリケーション格納部4103に格納されたアプリケーションの認証の処理を、T R Mアクセスライブラリ部4104により行なう。

【0211】

本実施の形態においては、認証モジュール4102のT R Mアクセスライブラリ部認証部4106により認証されたT R Mアクセスライブラリ部4104がアプリケーションの認証を行なう。このため、アプリケーションの認証の結果が、認証モジュール4102によって、信頼できるものとなる。この結果、アプリケーション格納部4103に格納されたアプリケーションの実行により、認証モジュールへのアクセスが発生した場合に、認証モジュールはそのアクセスを許可することができる。また、例えば、アプリケーション格納部4103が、他の機器へアプリケーションをダウンロードする目的で備えられたものであっても、アプリケーション格納部4103からダウンロードされたアプリケーションからの認証モジュールへのアクセスを許可することができる。また、アプリケーションに、認証モジュール4102により認証されたことを示す情報を付加することも可能である。これにより、従来の処理よりも複雑な処理を実現することが可能となる。

40

【0212】

(実施の形態20)

50

実施の形態 19 などにおいて示されたアプリケーション認証システムは、第一の機器と、認証モジュールと、の 2 つの主要な機器から構成されていた。しかし、本実施の形態において示されるように、主要な機器の数は、2 つに限定されるものではない。

【0213】

図 43 は、主要な機器の数を 3 とした場合のアプリケーション認証システムの機能ブロック図を例示している。このアプリケーション認証システムは、第 1 の機器 4301 と、第 2 の機器 4302 と、第 3 の機器 4303 と、からなる。すなわち、アプリケーション認証システムは、3 つの主要な部分を持つ。なお、これらの 3 つの主要な部分は直列に接続されているが、実施の形態 18 や実施の形態 19 で述べたように、電氣的に直接接続されていたり、物理的に接触されるようになっていたりする必要はない。また、第 1 の機器 4301、第 2 の機器 4302、第 3 の機器 4303 を所有あるいは占有している者が、一人であってもよいし、これら 3 つの機器が、異なる者により所有あるいは占有されていてもよい。

10

【0214】

実施の形態 19 などのアプリケーション認証システムとの対応は次の通りである。すなわち、第 1 の機器 4301 が認証モジュールに相当し、第 3 の機器 4303 が、端末や実施の形態 19 における第一の機器 4101 に相当する。

【0215】

第 1 の機器 4301 は、TRM 部 4304 と、第 1 認証処理部 4305 と、を有している。

20

【0216】

TRM 部 4304 は、第 2 の機器を認証するための認証情報を耐タンパ領域に保持する。したがって、実施の形態 19 などの TRM 部と同じ機能を有する。ただ、耐タンパ領域に保持される情報が第 2 の機器を認証するための認証情報となっている点が異なる。なお、耐タンパ領域には、第 2 の機器を認証するための情報だけでなく、第 3 の機器を認証するための情報も保持されていてもよい。この認証情報としては、実施の形態 19 での TRM アクセスライブラリ部認証情報と同様に、機器の製造番号や、機器の置かれている状況を示す情報、あるいは、機器が暗号鍵や証明書を保持できる場合には、その暗号鍵や証明書の真正性を検出するための情報を挙げることができる。

【0217】

第 1 認証処理部 4305 は、前記認証情報に基づいて第 2 の機器 4302 を認証する。「前記認証情報」とは、耐タンパ領域に保持された認証情報である。

30

【0218】

第 2 の機器 4302 は、第 2 認証処理部 4306 を有する。第 2 認証処理部 4306 は、第 1 認証処理部に自身が認証されることを条件として第 3 の機器 4303 を認証する。ここに「自身」とは、第 2 認証処理部 4306 を含む第 2 の機器 4302 を意味する。

【0219】

第 2 の機器 4302 が第 1 認証処理部 4305 により認証された場合、第 2 の機器 4302 は第 1 の機器 4301 にとって信頼できる機器とみなすことができる。そのため、第 1 の機器 4301 は、耐タンパ領域に保持される情報へ第 2 の機器 4302 がアクセスすることを許すことができる。そこで、第 2 認証処理部 4306 が第 3 の機器 4303 を認証する場合には、耐タンパ領域に保持された情報を用いることが可能である。そこで、第 2 認証処理部 4306 が第 3 の機器 4303 を認証する際には、耐タンパ領域に保持された情報を用いるようにしてもよい。

40

【0220】

このように、第 2 認証処理部 4306 は、第 1 の機器 4301 へアクセスするためのデバイスドライバの機能を備えていてもよい。

【0221】

また、第 2 認証処理部 4306 は、第 3 の機器 4303 を認証するために必要な情報を、第 1 の機器や第 3 の機器とは異なる別の機器より取得するようにしてもよい。この取得の

50

際に、第2認証処理部4306は、第2の機器4302が第1認証処理部4305により認証されていることを示す情報を提示するようにしてもよい。

【0222】

第3の機器4303は、アプリケーション格納部4307と、第3認証処理部4308と、を有している。アプリケーション格納部4307は、アプリケーションを格納する。したがって、実施の形態19などにおけるアプリケーション格納部と同じ機能を有する。このため、アプリケーションを格納する時間の長さや格納する目的は特に限定されない。

【0223】

第3認証処理部4308は、第2の機器4302に自身が認証されることを条件として、前記アプリケーションの認証のための処理をする。「前記アプリケーション」とは、アプリケーション格納部4307に格納されているアプリケーションを意味する。また、「自身」とは、第3の機器4303を意味する。

10

【0224】

第3の機器4303が第2認証処理部4306により認証された場合、第3の機器4303は第1の機器4301にとって信頼できる機器とみなすことができる。そのため、第1の機器4301は、耐タンパ領域に保持される情報へ第3の機器4303がアクセスすることを許すことができる。そこで、第3認証処理部4306がアプリケーションを認証する場合には、耐タンパ領域に保持された情報を用いることが可能である。

【0225】

また、第3認証処理部4308は、アプリケーションを認証するために必要な情報を、第1の機器や第2の機器とは異なる別の機器より取得するようにしてもよい。この取得の際に、第3認証処理部4308は、第3の機器4302が第2認証処理部4306により認証されていることを示す情報を提示するようにしてもよい。また、その際、第2の機器4302が第1認証処理部4305により認証されていることを示す情報を提示してもよい。

20

【0226】

図44は、第1の機器4301の動作を説明するフローチャートを例示する。ステップS4401において、第1認証処理部4305はTRM部4304が耐タンパ領域に保持する認証情報を取得する。ステップS4402において、第2の機器の認証の処理を行ない、ステップS4403において、認証できたかどうかを判断する。もし、認証できたのであれば、ステップS4404において、第2の機器へ、認証されたことを伝達する。

30

【0227】

図45は、第2の機器4302の動作を説明するフローチャートを例示する。ステップS4501において、第1認証処理部4305により認証されたかどうかを判断する。この判断は、図44に示されたステップS4404において、認証されたことが第1の機器4301より伝達されたかどうかにより判断される。もし、認証されたのであれば、ステップS4502において、第3の機器の認証の処理を、第2認証処理部4306により行なう。ステップS4503において、認証できたかどうかを判断する。もし、認証できたのであれば、ステップS4504において、第3の機器へ認証されたことを伝達する。

【0228】

図46は、第3の機器4303の動作を説明するフローチャートを例示する。ステップS4601において、第2認証処理部4306により認証されたかどうかを判断する。この判断は、図45に示されたステップS4504において、認証されたことが第2の機器4302から伝達されたかどうかにより判断される。もし、認証されたのであれば、ステップS4602において、アプリケーションの認証の処理を行なう。図46では、ここで処理が終了するが、アプリケーションの認証の処理の結果を、例えば、第1の機器4301へ伝達してもよい。また、第2の機器4302へ伝達してもよい。また、もし、アプリケーションが他の機器へアクセスする場合には、第3認証処理部4308により認証されたことを示す情報を提示するようにしてもよい。あるいは、アプリケーションに第3認証処理部4308により認証されたことを示す情報を付加してもよい。

40

50

【 0 2 2 9 】

以上は、3つの主要な機器によりアプリケーション認証システムが構成される場合であったが、図47に例示されるように、アプリケーション認証システムは、4つの主要な機器により構成されていてもよい。

【 0 2 3 0 】

図47においては、アプリケーション認証システムは、第1の機器4701と、第2の機器4702と、第3の機器4703と、第4の機器4704と、により構成される。第1の機器4701は、TRM部4705と、第1認証処理部4706と、を有しており、これらは、図43に例示されたTRM部4304と、第1認証処理部4305と、に対応する。第2の機器4702は、第2認証処理部4707を有している。これは、図43に例示された第2認証処理部4306に対応する。

10

【 0 2 3 1 】

第3の機器は、第3認証処理部4708を有している。第3認証処理部4708は、第2認証処理部4707に自身が認証されることを条件として第4の機器4704を認証する。第2認証処理部4707により第3の機器4703が認証されれば、第3の機器4703は、第1の機器4701にとって信頼できる機器となるので、第1の機器4701は、第3の機器4703が、TRM部4705が耐タンパ領域に保持する情報にアクセスすることを許可することができる。そこで、第3認証処理部4708は、TRM部4705が耐タンパ領域に保持する情報を取得し、第4の機器4704を認証するようにしてもよい。また、他の機器に保持されている情報を取得し、認証するようにしてもよい。

20

【 0 2 3 2 】

なお、第3認証処理部4708が、TRM部4705が耐タンパ領域に保持する情報を取得する場合には、第3の機器4703と第1の機器4701とが直接通信を行なえるようにするために、第3の機器4703と第1の機器4701とに通信部が備えられていてもよい。あるいは、第2認証処理部4707が第1の機器4701へアクセスするためのデバイスドライバの機能を備えている場合には、次のようにしてもよい。すなわち、第3認証処理部4708は、第2認証処理部4707の備えるデバイスドライバの機能を用いて、第2の機器4702を介しながら、TRM部4705が耐タンパ領域に保持する情報を取得する。したがって、第3認証処理部4708は、第2認証処理部4707へアクセスするためのデバイスドライバの機能を備えていてもよい。

30

【 0 2 3 3 】

第4の機器4704は、アプリケーション格納部4709と、第4認証処理部4710と、を有する。アプリケーション格納部4709は、図43のアプリケーション格納部4307に対応する。第4認証処理部4710は、第3認証処理部4708に自身が認証されることを条件として、アプリケーションを認証する。ここで、「自身」とは、第4の機器を意味している。したがって、第3認証処理部4708により第4の機器が認証されていれば、第1認証処理部4706により認証されたものとみなされるので、第4認証処理部は、TRM部4705が耐タンパ領域に保持する情報を用いて、アプリケーション格納部4709に格納されているアプリケーションの認証をすることができる。また、第4認証処理部4710は、第3認証処理部4708により認証されたことを示す情報を、他の機器に提示して、アプリケーションの認証のための情報を取得して、アプリケーションの認証を行なうようにしてもよい。

40

【 0 2 3 4 】

このように、4つの主要な機器によりアプリケーション認証システムが構成される場合の動作は次のようになる。第1の機器4701の動作は、図44に例示されたものと同じである。同様に第2の機器4702の動作は、図45に例示されている。第3の機器4703の動作は、図48に例示されているものとなる。すなわち、第3の機器4703が第2認証処理部4707により認証されたかどうかを、ステップS4801において、判断する。もし、認証されたのであれば、ステップS4802において、第4の機器4704の認証の処理を行なう。ステップS4803において、認証ができたかどうかを判断し、も

50

し、認証できたのであれば、ステップ S 4 8 0 4 において、第 4 の機器 4 7 0 4 へ認証されたことを伝達する。

【 0 2 3 5 】

第 4 の機器の動作は、図 4 9 に例示されている。第 3 認証処理部 4 7 0 8 により認証されたかどうかを、ステップ S 4 9 0 1 において、判断し、もし、認証されたのであれば、ステップ S 4 9 0 2 において、アプリケーションの認証の処理を行なう。このアプリケーションの認証の処理の結果を、例えば、第 1 の機器 4 7 0 1 へ伝達してもよい。また、第 2 の機器 4 7 0 2 へも伝達してもよい。また、もし、アプリケーションが他の機器へアクセスする場合には、第 3 認証処理部 4 7 0 8 により認証されたことを示す情報を提示するようにしてもよい。あるいは、アプリケーションに第 3 認証処理部 4 7 0 8 により認証されたことを示す情報を付加してもよい。

10

【 0 2 3 6 】

更に、主要な機器の数は 4 に限定されることなく、図 5 0 に例示されるように、第 1 の機器 5 0 0 1 と、第 2 の機器 5 0 0 2 と、第 3 の機器 5 0 0 3 と、第 4 の機器 5 0 0 4 と、第 5 の機器 5 0 0 5 と、の 5 つの機器によりアプリケーション認証システムが構成されていてもよい。

【 0 2 3 7 】

図 5 1 は、主要な数を一般化した場合のアプリケーション認証システムの機能ブロック図を例示する。図 5 1 において、アプリケーション認証システムは、第 1 の機器 5 1 0 1 から第 (N + 1) の機器 5 1 0 5 までを直列に接続してできる (N + 1) 個の機器により構成される。「接続」と書いたが、これは、電氣的に直接接続されていること、あるいは、物理的に接触していることだけを意味しているのではない。例えば、インターネットに代表されるようなネットワークによって接続されていてもよい。特に、光ケーブルを用いたネットワークによって接続されていてもよい。また、無線によって接続されていたりしてもよい。また、それぞれの機器が相互に接続されていてもよい。

20

【 0 2 3 8 】

第 1 の機器 5 1 0 1 は、T R M 部 5 1 0 1 と第 1 認証処理部 5 1 0 2 と、を有している。T R M 部 5 1 0 1 は、第 2 の機器 5 1 0 2 を認証するための情報である認証情報を耐タンパ領域に保持する。第 1 認証処理部 5 1 0 2 は、認証情報に基づいて第 2 の機器 5 1 0 2 を認証する。

30

【 0 2 3 9 】

以下では、第 2 の機器 5 1 0 2 から第 N の機器までのいずれか一の機器を第 i の機器と表記することにする。第 i の機器は、第 i 認証処理部を有する。第 i 認証処理部は、第 (i - 1) 認証処理部に自身が認証されることを条件として第 (i + 1) の機器を認証する。ここに「自身」とは、第 i の機器を意味する。したがって、第 i 認証処理部は、第 i の機器が第 (i - 1) 認証処理部に認証されているならば、第 (i + 1) の機器を認証する。第 i の機器が第 (i - 1) 認証処理部に認証されているならば、第 1 の機器 5 1 0 1 によって、第 i の機器は信頼できるとみなすことができるので、第 1 の機器 5 1 0 1 は、第 i の機器が、T R M 部 5 1 0 1 により耐タンパ領域に保持されている情報にアクセスを許すことができる。そこで、第 i の機器は、耐タンパ領域に保持された情報を用いて、第 (i + 1) の機器を認証するようにしてもよい。

40

【 0 2 4 0 】

例えば、第 i 認証処理部が、T R M 部 5 1 0 7 により耐タンパ領域に保持されている情報にアクセスする場合には、第 i の機器と第 1 の機器とが直接通信を行なうようにしてもよい。

【 0 2 4 1 】

また、第 i 認証処理部は、第 (i - 1) 認証処理部へアクセスするためのデバイスドライバなどの機能を備えていてもよい。この備えられた機能により、情報の要求が第 i 認証処理部から第 1 認証処理部 5 1 0 7 へ順に出力され、要求された情報が第 1 認証処理部 5 1 0 7 から第 i 認証処理部へ順に返信されるようになる。

50

【0242】

第(N+1)の機器5105は、アプリケーション格納部5111と、第(N+1)認証処理部5112と、を有している。アプリケーション格納部5111は、アプリケーションを格納する。アプリケーションを格納している時間は特に問わず、例えば、第(N+1)の機器5105にてアプリケーションを実行するためにアプリケーションが格納されるようになっていてもよいし、アプリケーションの伝達を中継するためだけに一時的に格納されるようになっていてもよい。また、格納の目的も特に問わず、第(N+1)の機器5105にてアプリケーションを実行する場合のみならず、他の機器でのアプリケーションの実行のためのダウンロードの目的に格納してもよい。また、第(N+1)の機器5105でアプリケーションをダウンロードして実行する目的で格納してもよい。なお、アプリケーションという名称を用いたが、プログラムである必要はなく、データであってもよい。

10

【0243】

第(N+1)認証処理部5112は、第N認証処理部に自身が認証されることを条件として、アプリケーションを認証する。「自身」とは、第(N+1)の機器5105を意味し、「アプリケーション」とは、アプリケーション格納部5111に格納されたアプリケーションを意味する。

【0244】

第1の機器5101の動作を説明するフローチャートは、図44に例示されている。

【0245】

20

図52は、第2の機器から第Nの機器までのいずれか一の機器の動作を説明するフローチャートである。ステップS5201において、第(i-1)認証処理部により認証されたかどうかを判断し、もし、認証されれば、第(i+1)の機器の認証の処理をステップS5202において行なう。第(i+1)の機器が認証されたかどうかを、ステップS5203において、判断し、もし、認証されたのであれば、第(i+1)の機器へ認証されたことを伝達する。これにより、第2の機器から順に認証が第Nの機器までが認証されることになる。

【0246】

図53は、第(N+1)の機器5105の動作のフローチャートを例示している。第N認証処理部5105により認証されたかどうかをステップS5301において判断し、もし認証されたと判断されれば、ステップS5302において、アプリケーションの認証の処理を行なう。アプリケーションの認証ができれば、その結果を第1の機器5101などへ伝達し、アプリケーションによる第1の機器5101へのアクセスすることを許可するようにする。あるいは、認証がされたということを示す情報をアプリケーションに付加するようにしてもよい。

30

【0247】

図43、図47、図50、図51においては、機器が直列に接続されていたが、入れ子構造を用いて機器が接続されていてもよい。図54は、入れ子構造を用いて機器が接続されていることを模式的に例示している。セキュアデバイス5401が第1の機器に相当し、TRM部と第1認証処理部とを有する。セキュアデバイス5402は、第2の機器に相当するアダプタ5402の一部として組み込まれるように接続される。アダプタ5402は、第3の機器に相当する通信モジュール5403の一部として組み込まれるように接続されている。さらに通信モジュール5403は、第4の機器に相当するPDA(Personal Digital Assistance)5404の一部として組み込まれるように接続される。PDA5404は、サーバ5405と通信回線などを通じて第5の機器に相当するサーバに接続される。このように接続が行なわれることにより、例えば、サーバ5405に格納されたアプリケーションを、セキュアデバイス5401の耐タンパ領域に保持された情報を用いて認証することができる。

40

【0248】

なお、第(N+1)の機器5105は、例えば、家庭用電化製品であってもよい。また、

50

いわゆる、情報家電やネット家電と呼ばれるものであってもよい。そのような家庭電化製品を例示列挙すれば、エアコンディショナ、加湿器、除湿器、空気清浄機、電子レンジ、オーブン、冷蔵庫、食器洗い機、湯沸し器、アイロン、ズボンプレスサー、電気掃除機、洗濯機、乾燥機、電気毛布、電気敷布、照明機器、テレビ受像機、ラジオ受信機、テーブルレコーダなどのオーディオ機器、カメラ、ＩＣレコーダ、電話機、ファクシミリ送受信機、コピー機、プリンター、スキャナー、パーソナルコンピュータ、などを挙げることができる。

【 0 2 4 9 】

(実施の形態 2 1)

実施の形態 2 0 においては、第 2 の機器から第 N の機器までのいずれか一を第 i の機器と表記するとき、第 i の機器は、第 $(i + 1)$ の機器の認証を行なうために必要な情報を保持するようになっていてもよい。例えば、第 i の機器が耐タンパ領域を有し、そこに認証を行なうのに必要な情報を保持していてもよい。

10

【 0 2 5 0 】

このように第 i の機器が認証を行なうために必要な情報である認証情報を保持している場合があり得るとき、第 i の機器が第 i 認証情報取得部を有していてもよい。ここに、第 i 認証情報取得部とは、第 i の機器が認証情報を格納している領域である認証情報格納領域を有している場合には、その認証情報格納領域から認証情報を取得する部である。また、第 i 認証情報取得部は、第 1 の機器の T R M 部に認証情報が格納されている場合には、第 2 の機器から第 $(i - 1)$ の機器までの機器を介して、T R M 部から認証情報を取得する。「第 2 の機器から第 $(i - 1)$ の機器までの機器を介して」とは、次を意味する。すなわち、第 i 認証情報取得部が、第 $(i - 1)$ 認証情報取得部へ認証情報の要求を出力し、最後には、第 2 認証情報取得部が第 1 の機器へ認証情報の要求を出力する。第 2 認証情報取得部が第 1 の機器から認証情報を取得すると、第 3 認証情報取得部へ認証情報を出力し、最後には、第 $(i - 1)$ 認証情報取得部が第 i 認証情報取得部へ認証情報を出力する。

20

【 0 2 5 1 】

図 5 5 は、本実施の形態における第 i の機器の動作を説明するフローチャートを例示する。ステップ S 5 5 0 1 において、第 $(i - 1)$ 認証処理部により認証されたかどうかを判断する。認証されたと判断されれば、ステップ S 5 5 0 2 へ処理が移され、認証情報格納領域に、第 $(i + 1)$ の機器の認証情報が格納されているかどうかを判断する。例えば、認証情報格納領域に格納されている情報を読み出し、認証情報が得られるかどうかを判断する。もし、格納されていなければ、ステップ S 5 5 0 3 へ処理が移され、第 2 の機器から第 $(i - 1)$ の機器までを介して T R M 部より認証情報を取得する。ステップ S 5 5 0 4 において、第 $(i + 1)$ の機器の認証の処理を行ない、認証できたかどうかをステップ S 5 5 0 5 において判断する。もし、認証できたのであれば、ステップ S 5 5 0 6 において、第 $(i + 1)$ の機器へ認証されたことを伝達する。

30

【 0 2 5 2 】

このような処理を行なうことにより、アプリケーション認証システムを構成する機器の数が増えても、機器が認証情報を持っていれば、その認証情報を使用して認証がされるので、例えば、全ての機器の認証の時間を短時間に終わらせることが可能となる。

40

【 0 2 5 3 】

【 発明の効果 】

以上のように本発明によれば、認証モジュールと端末とを組み合わせ、端末内で動作するアプリケーションを認証モジュールの持つ情報で認証することにより、好ましくないアプリケーションが端末内で実行されてしまうことを防止することができる。

【 0 2 5 4 】

また、端末内の T R M アクセスライブラリが認証モジュールによって認証されることにより、T R M アクセスライブラリがアプリケーションの認証のための処理を行なうことが可能となる。この結果、高いセキュリティを伴うアプリケーションの実行が可能となり、端末を用いた商取引の操作が可能となる。また、耐タンパ領域は認証モジュールにあればよ

50

く、端末に耐タンパ領域を実装する必要がなくなるので、端末の製造コストを下げることができる。

【 0 2 5 5 】

また、端末内で動作するアプリケーションが認証されることにより、アプリケーションの出所が保証され、端末や認証モジュールのローカルリソースのアクセスをアプリケーションに許可することができる。

【 0 2 5 6 】

また、サーバが認証モジュールの T R M 部を認証することにより、端末を認証すること、および、端末内で動作するアプリケーションの認証を行なうことができるので、機密性の高い情報を端末内で動作するアプリケーションと交換することができ、複雑な商取引の操作をサーバ、端末、認証モジュール間で実現することができる。

10

【 0 2 5 7 】

また、オプショナル領域にアプリケーション本体の署名を格納することにより、アプリケーションのダウンロードと同時にアプリケーション本体の署名をダウンロードすることが可能となり、別にアプリケーション本体の署名をダウンロードする手間を省くことができる。

【 0 2 5 8 】

また、3つ以上の機器が直列に接続された状態で、一方の端の機器から順次認証を行ない、他の端の機器に格納されたアプリケーションの認証が可能となる。

【図面の簡単な説明】

20

【図 1】実施の形態 1 におけるアプリケーション認証システムの機能ブロック図

【図 2】実施の形態 1 におけるアプリケーション認証システムの実施例を示す図

【図 3】実施の形態 1 におけるアプリケーション認証システムの実施例を示す図

【図 4】実施の形態 2 におけるアプリケーション認証システムの機能ブロック図

【図 5】実施の形態 2 におけるアプリケーション認証システムの機能ブロック図

【図 6】アプリケーション本体と署名との関係を説明する図

【図 7】実施の形態 2 における端末の動作を説明するフローチャート

【図 8】実施の形態 2 における認証モジュールの動作を説明するフローチャート

【図 9】実施の形態 3 における認証モジュールの機能ブロック図

【図 10】実施の形態 3 における認証モジュールの動作を説明するフローチャート

30

【図 11】実施の形態 4 におけるアプリケーション認証システムの機能ブロック図

【図 12】実施の形態 4 における認証モジュールを端末が認証する処理を説明するフローチャート

【図 13】実施の形態 5 におけるアプリケーション認証システムの機能ブロック図

【図 14】アプリケーション利用リソース情報を模式的に表す図

【図 15】アプリケーション利用リソース情報がアプリケーションと共にダウンロードされる状態を模式的に表す図

【図 16】実施の形態 6 におけるアプリケーション認証システムの機能ブロック図

【図 17】実施の形態 7 におけるアプリケーション認証システムの機能ブロック図

【図 18】実施の形態 8 におけるアプリケーション認証システムの機能ブロック図

40

【図 19】実施の形態 8 における端末の動作を説明するフローチャート

【図 20】実施の形態 8 における認証モジュールの動作を説明するフローチャート

【図 21】実施の形態 9 におけるアプリケーション認証システムの機能ブロック図

【図 22】実施の形態 9 における認証モジュールの動作を説明するフローチャート

【図 23】実施の形態 9 における認証モジュールの動作を説明するフローチャート

【図 24】実施の形態 9 におけるアプリケーション認証システムの機能ブロック図

【図 25】実施の形態 9 を IC カードにおいて実現する例を示す図

【図 26】実施の形態 9 を IC カードにおいて実現する例を示す図

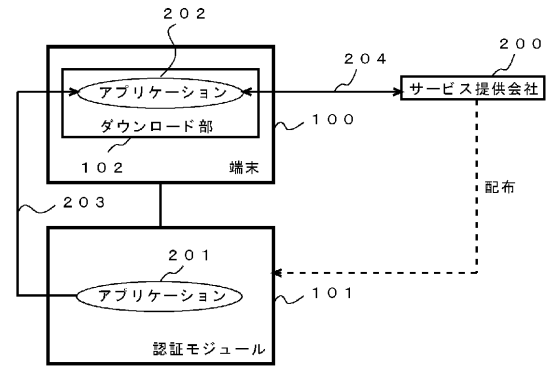
【図 27】実施の形態 10 におけるアプリケーション認証システムの機能ブロック図

【図 28】実施の形態 11 におけるアプリケーション認証システムの機能ブロック図

50

【図 29】実施の形態 12 におけるアプリケーション認証システムの機能ブロック図	
【図 30】実施の形態 12 におけるアプリケーション認証システムの動作を説明するシーケンス図	
【図 31】実施の形態 13 におけるアプリケーション認証システムの機能ブロック図	
【図 32】実施の形態 13 におけるアプリケーション認証システムの動作を説明するシーケンス図	
【図 33】実施の形態 14 におけるアプリケーション認証システムの機能ブロック図	
【図 34】J A V A (登録商標) アプリケーションのアプリケーション本体とアプリケーション定義ファイルを模式的に示す図	
【図 35】オプション領域に格納された署名を用いてアプリケーションの認証を行なう処理を説明するフローチャート	10
【図 36】実施の形態 2 のアプリケーション認証システムの T R M アクセスライブラリ部がアプリケーションマネージャとデバイスドライバとを備えている図	
【図 37】実施の形態 16 の端末の機能ブロック図	
【図 38】実施の形態 17 の端末の機能ブロック図	
【図 39】実施の形態 17 の端末における処理の流れを説明するフローチャート	
【図 40】実施の形態 18 におけるアプリケーション認証システムの機能ブロック図	
【図 41】実施の形態 19 におけるアプリケーション認証システムの機能ブロック図	
【図 42】実施の形態 19 におけるアプリケーション認証システムの処理の流れを説明するフローチャート	20
【図 43】3つの機器より構成されるアプリケーション認証システムの機能ブロック図	
【図 44】第 1 の機器の動作の流れを説明するフローチャート	
【図 45】第 2 の機器の動作の流れを説明するフローチャート	
【図 46】アプリケーション認証システムが 3つの機器より構成される場合における第 3 の機器の動作の流れを説明するフローチャート	
【図 47】4つの機器より構成されるアプリケーション認証システムの機能ブロック図	
【図 48】アプリケーション認証システムが 4つの機器より構成される場合における第 3 の機器の動作の流れを説明するフローチャート	
【図 49】アプリケーション認証システムが 4つの機器より構成される場合における第 4 の機器の動作の流れを説明するフローチャート	30
【図 50】5つの機器より構成されるアプリケーション認証システムの機能ブロック図	
【図 51】N + 1 の機器より構成されるアプリケーション認証システムの機能ブロック図	
【図 52】第 i の機器の動作の流れを説明するフローチャート	
【図 53】第 (N + 1) の機器の動作の流れを説明するフローチャート	
【図 54】入れ子構造を用いて機器が接続されていることを示す模式図	
【図 55】実施の形態 21 における第 i の機器の動作の流れを説明するフローチャート	
【符号の説明】	
100 端末	
101 認証モジュール	
102 ダウンロード部	40
103 T R M アクセス部	

【 図 2 】

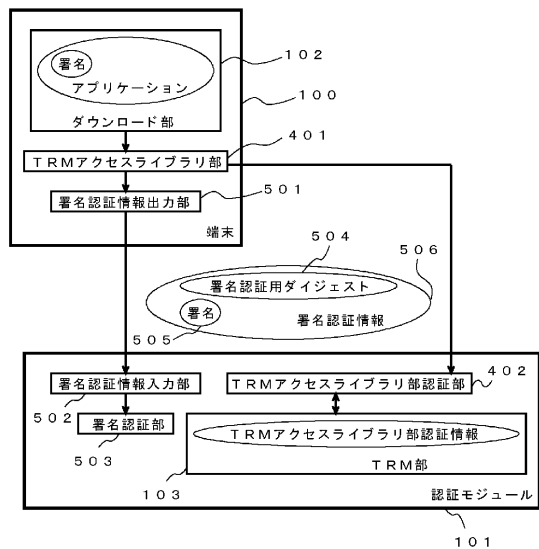


```

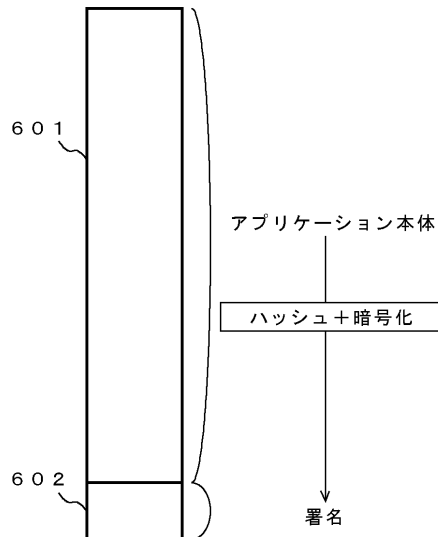
graph LR
    SP[サービス提供会社 200] -- 303 --> App2[アプリケーション 302]
    subgraph Terminal [端末 100]
        App2 <--> DU[ダウンロード部 102]
    end
    DU -- 304 --> App1[アプリケーション 301]
    subgraph AuthModule [認証モジュール 101]
        App1
    end
    App1 -- 305 --> App2

```

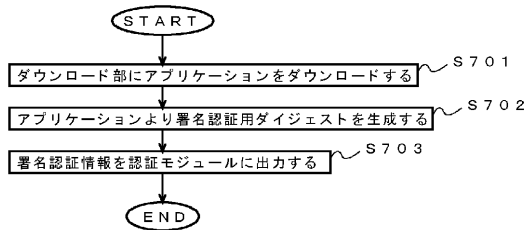
【 図 5 】



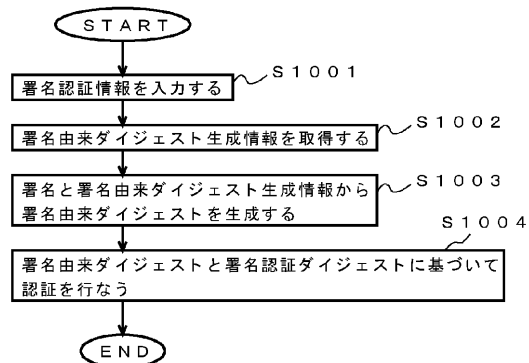
【図 6】



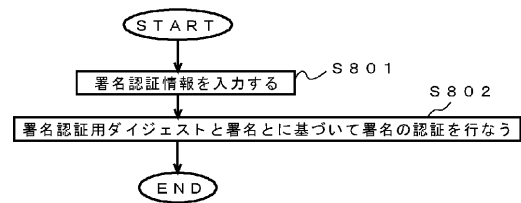
【図 7】



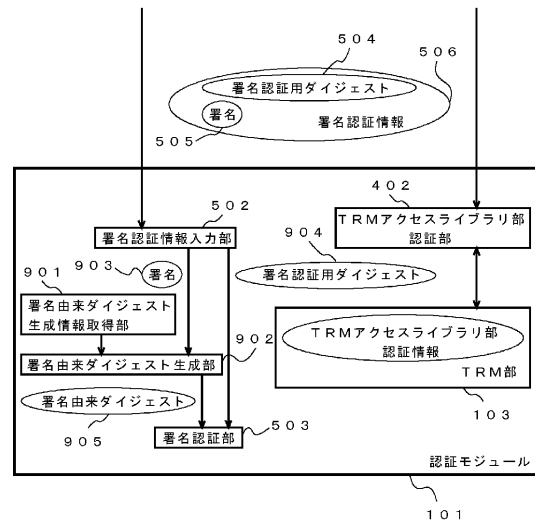
【図 10】



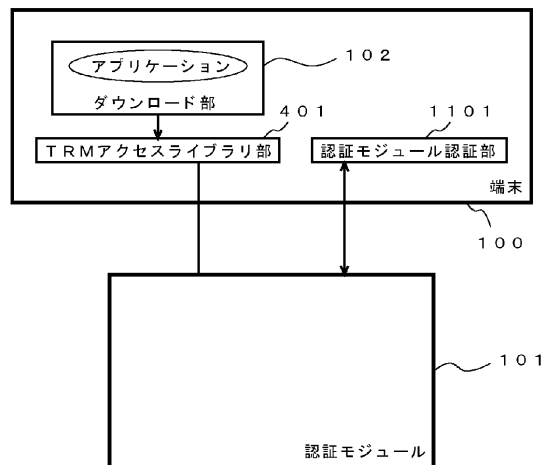
【図 8】



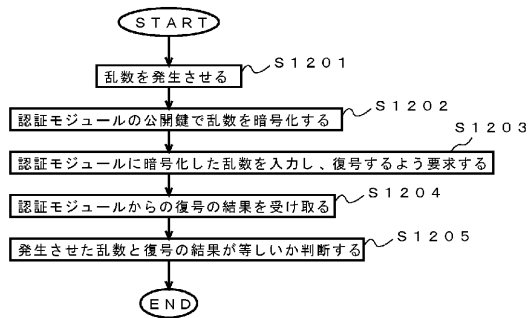
【図 9】



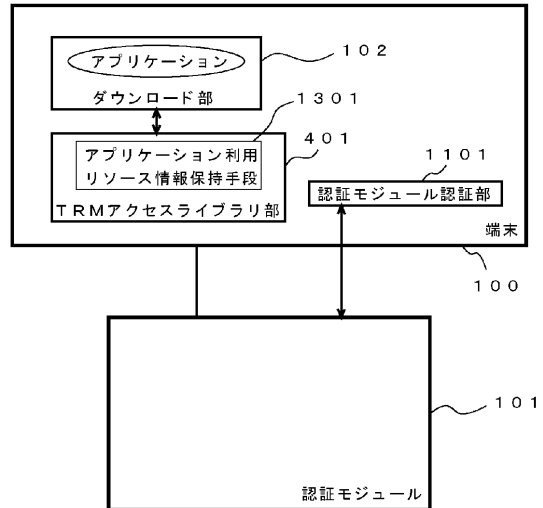
【図 11】



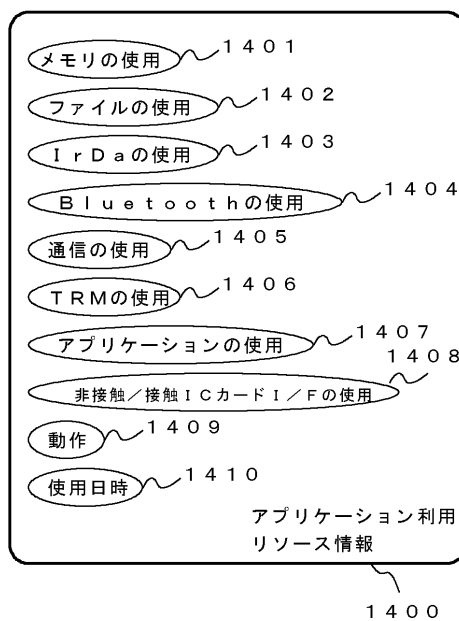
【図 12】



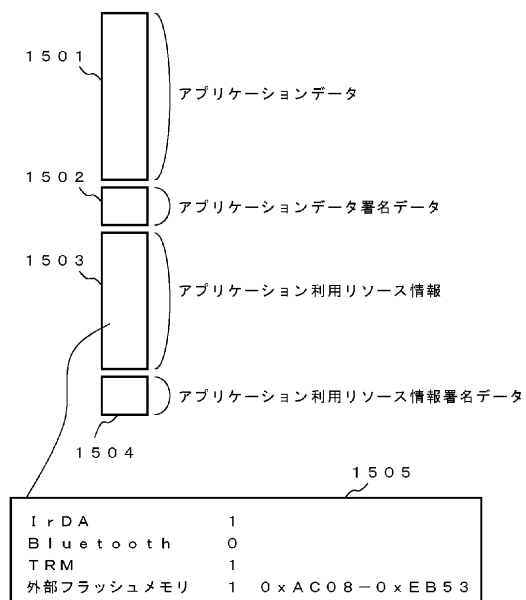
【図 13】



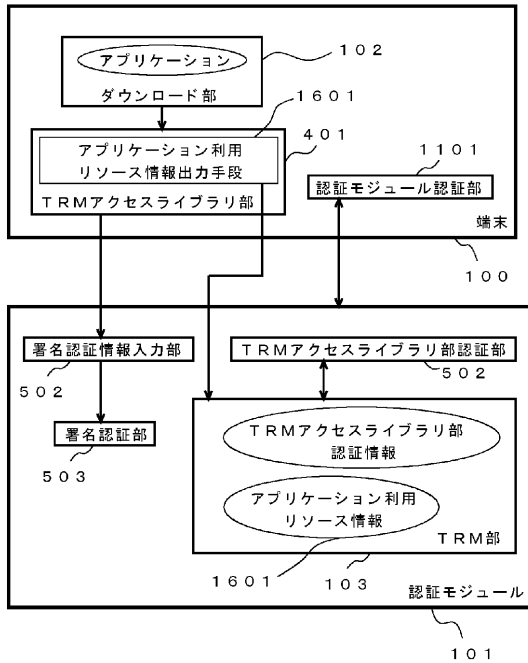
【図 14】



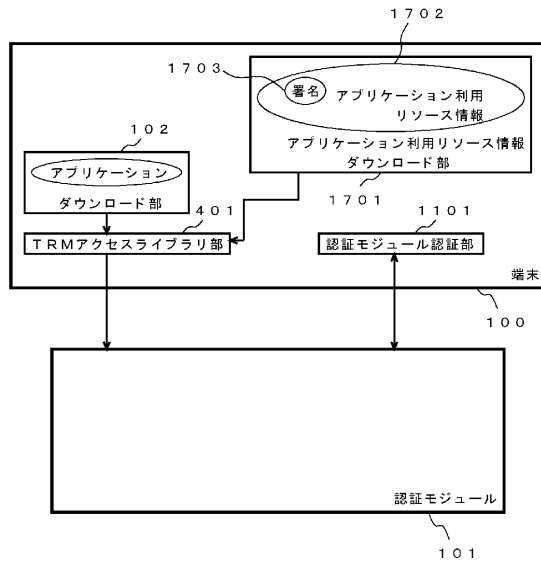
【図 15】



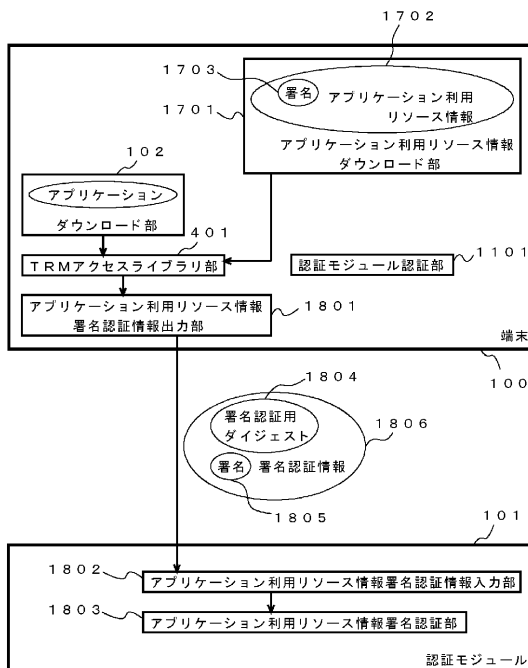
【図 16】



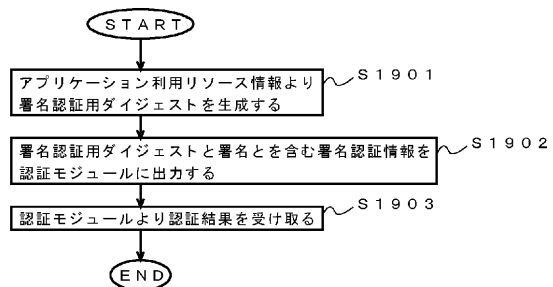
【図 17】



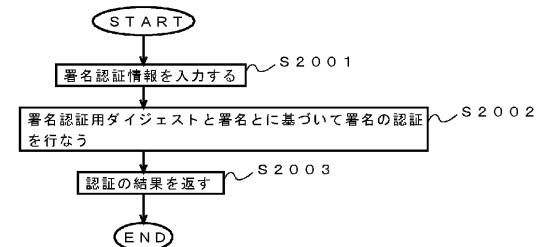
【図 18】



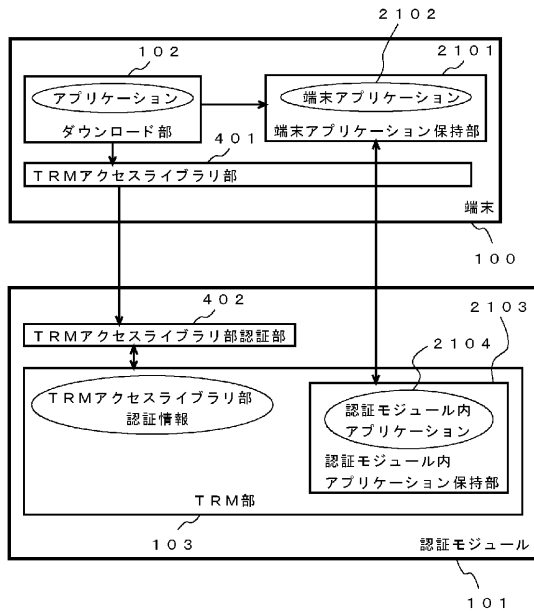
【図 19】



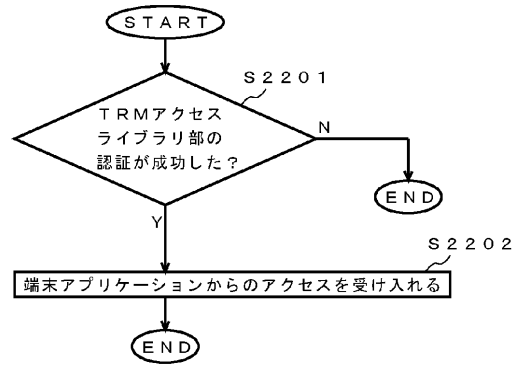
【図 20】



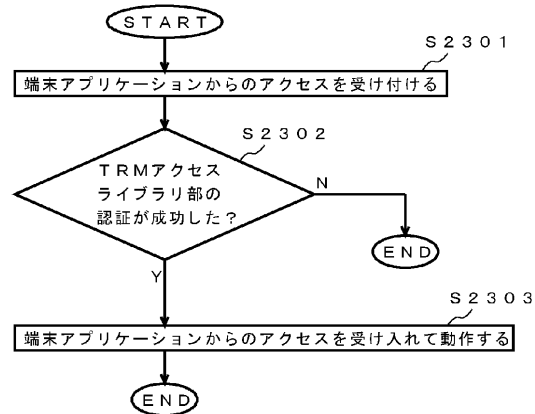
【図 21】



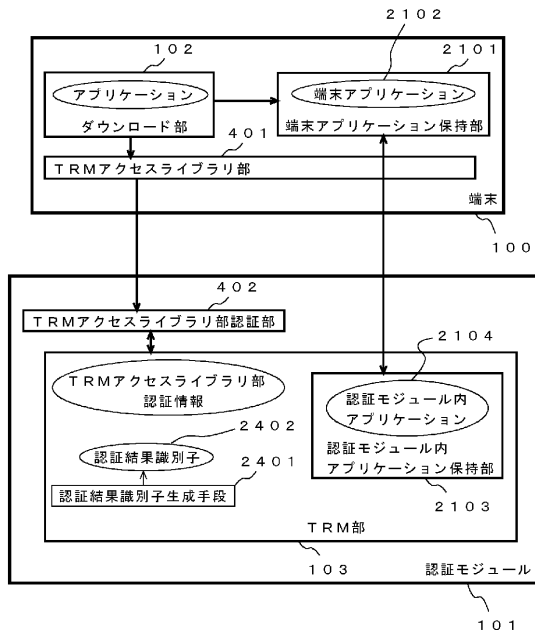
【図 22】



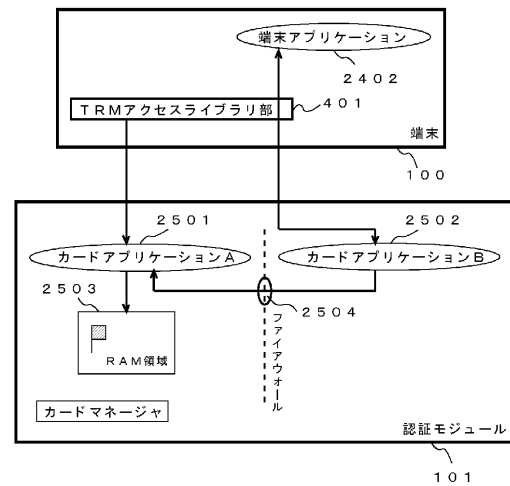
【図 23】



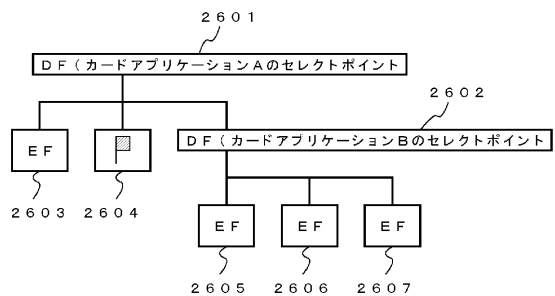
【図 24】



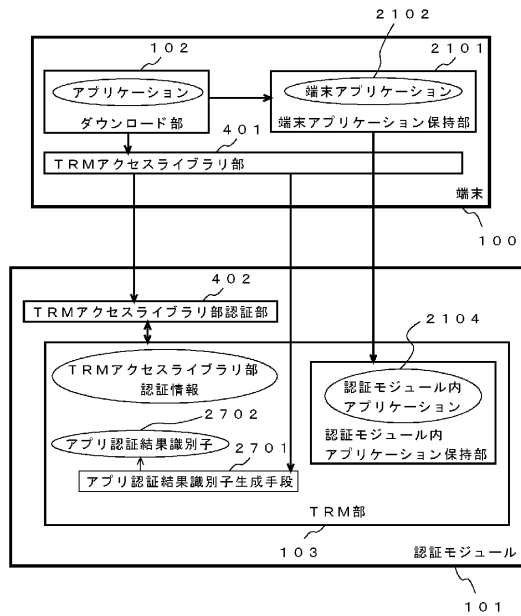
【図 25】



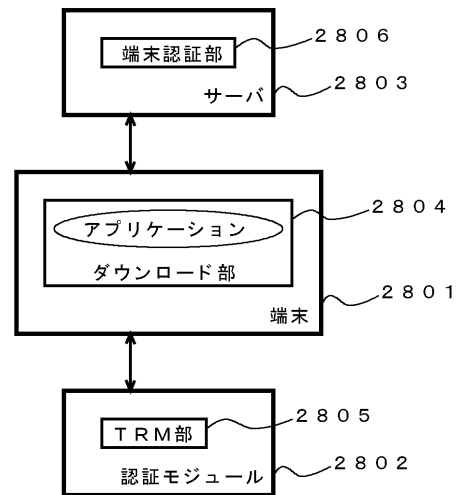
【図 26】



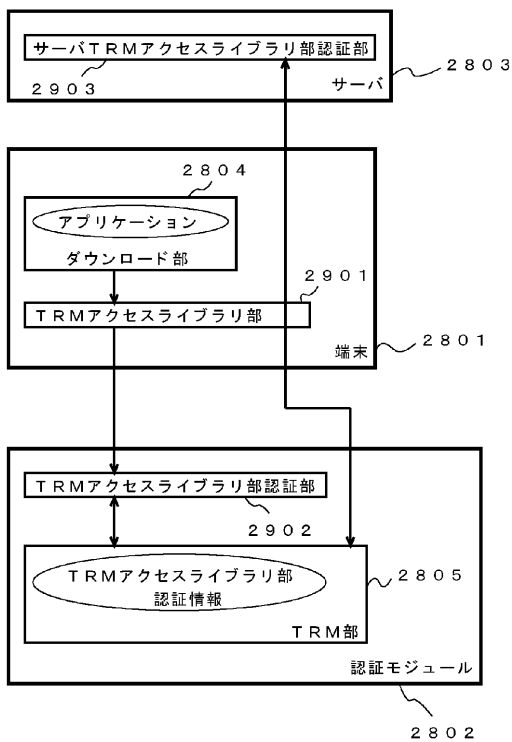
【図 27】



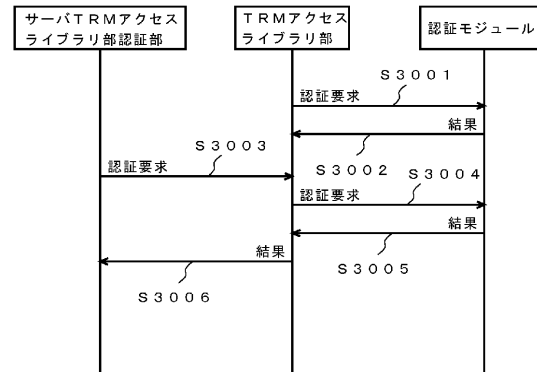
【図 28】



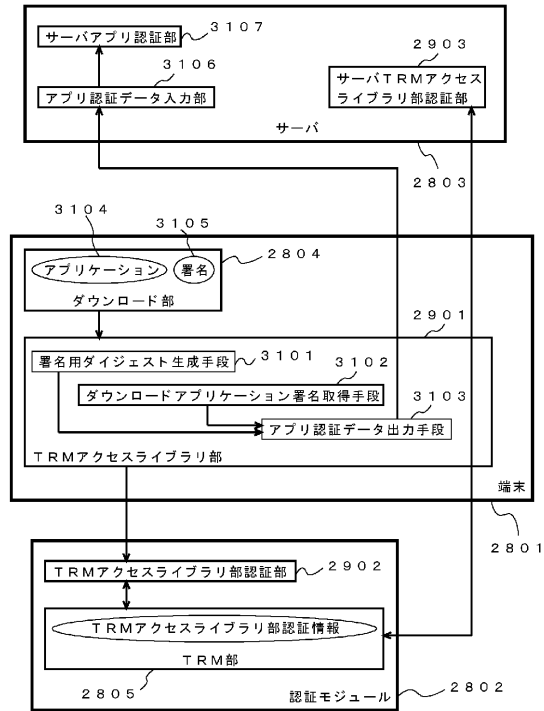
【図 29】



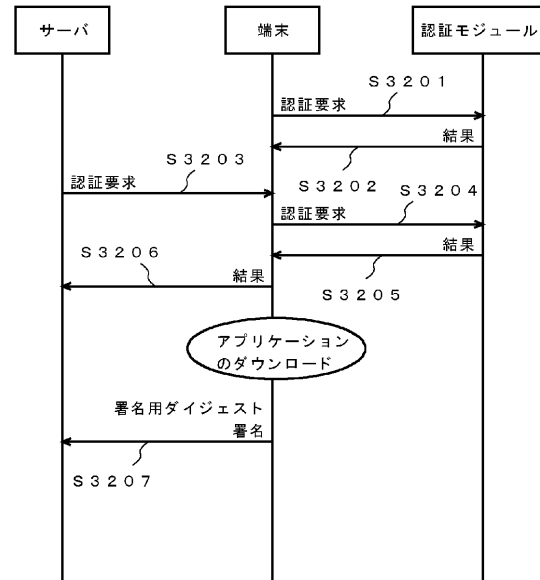
【図 30】



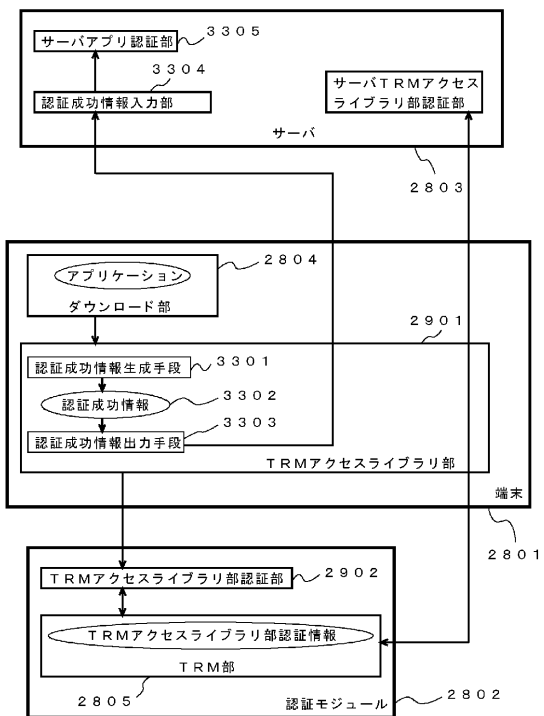
【図 3 1】



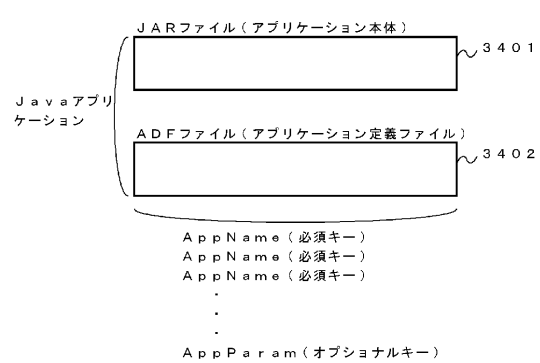
【図 3 2】



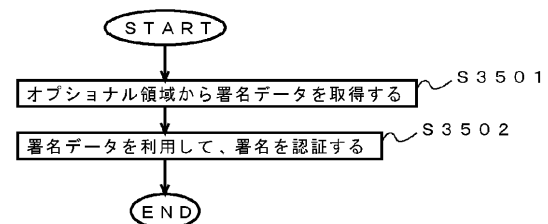
【図 3 3】



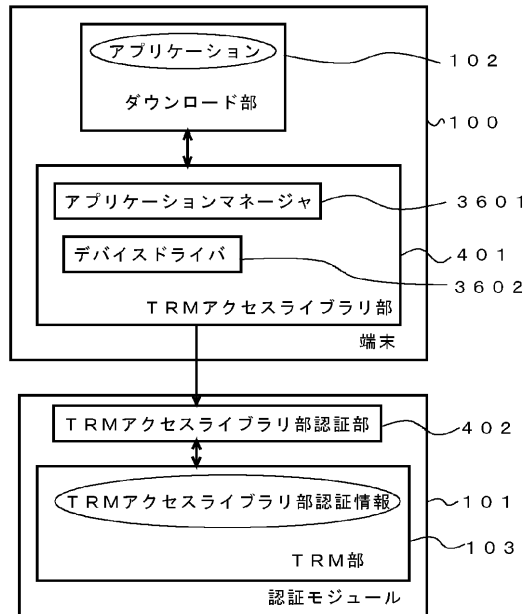
【図 3 4】



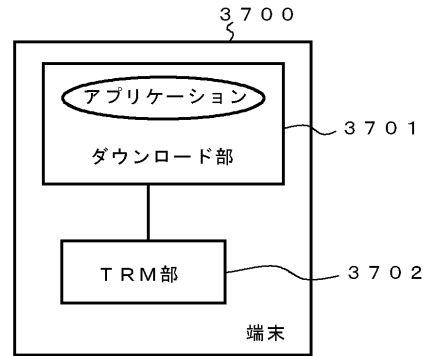
【図 3 5】



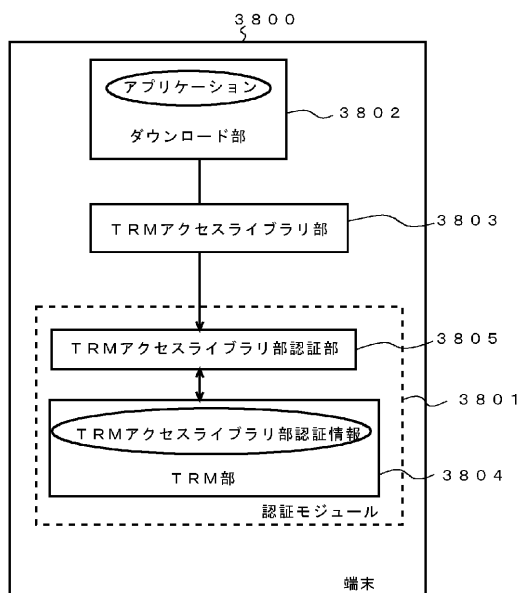
【図 36】



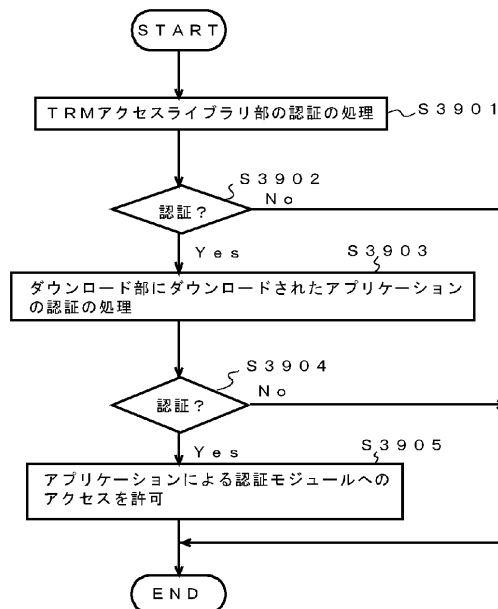
【図 37】



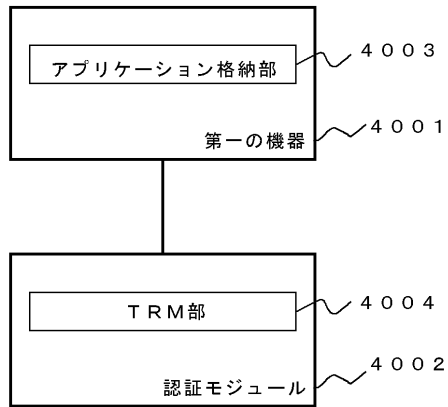
【図 38】



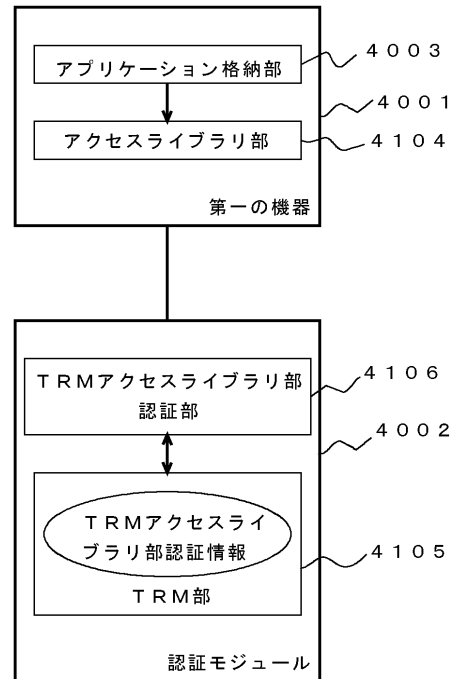
【図 39】



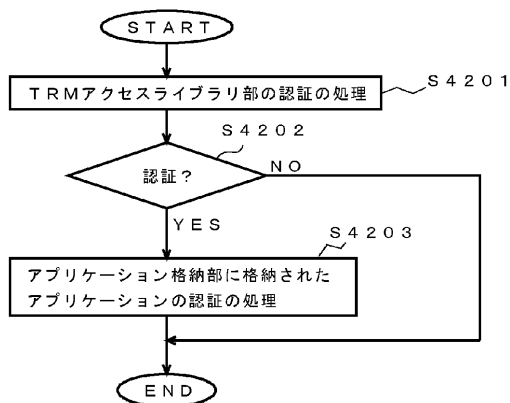
【図 40】



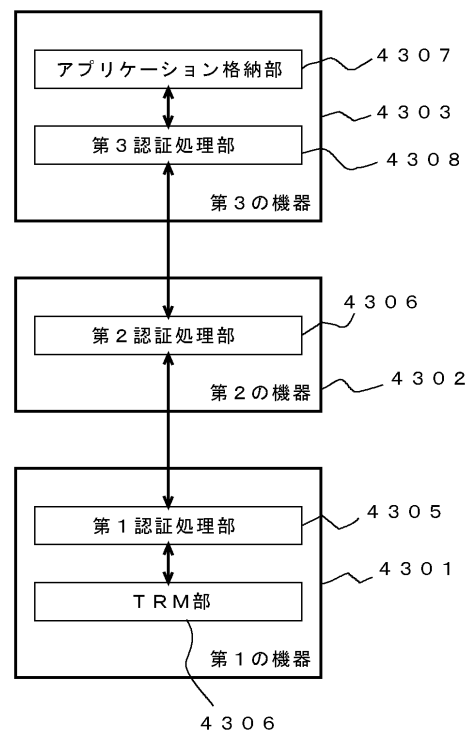
【図 41】



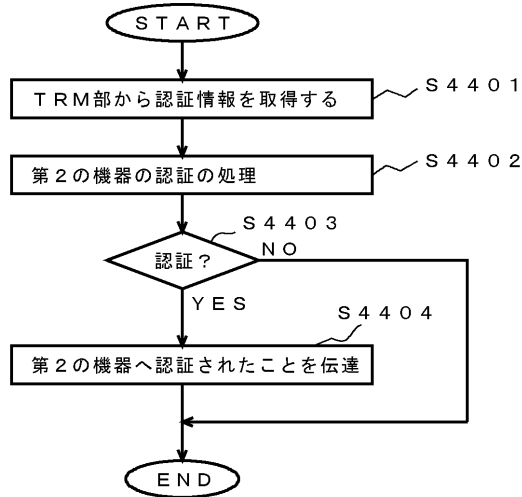
【図 42】



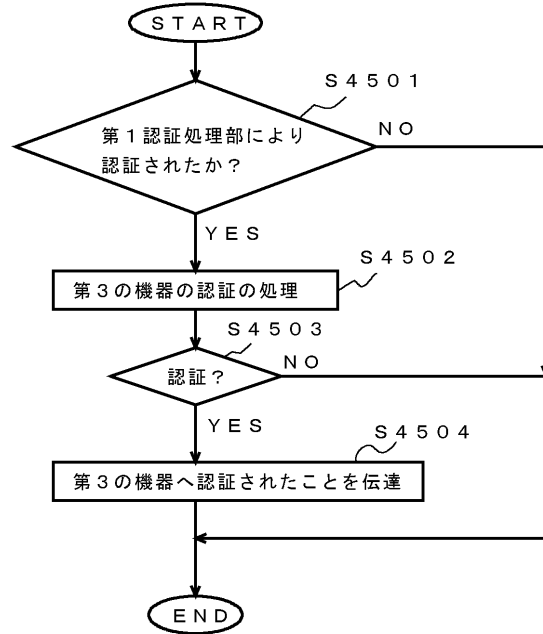
【図 43】



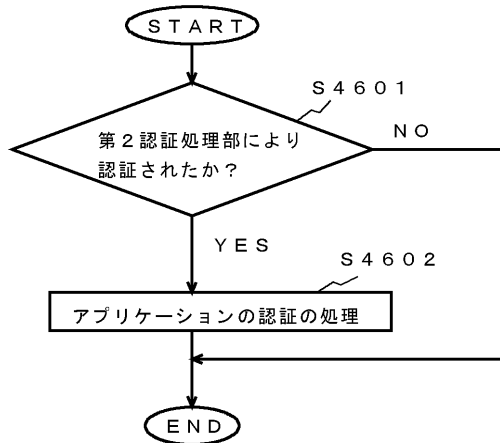
【図 4 4】



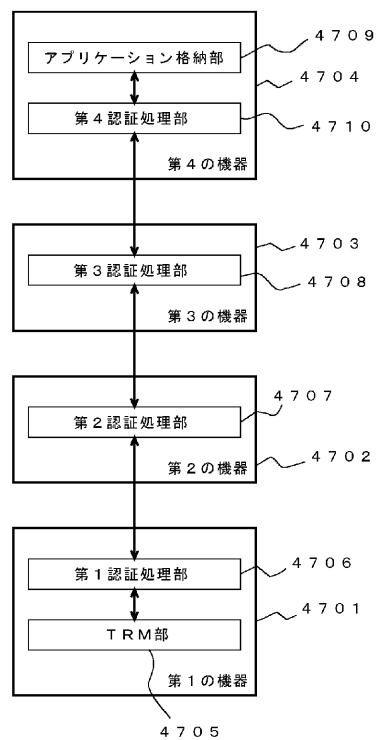
【図 4 5】



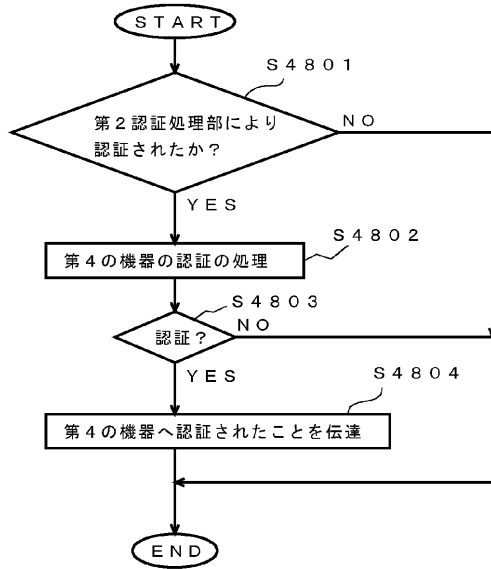
【図 4 6】



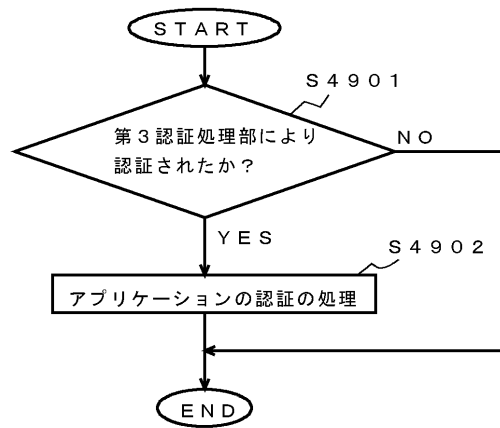
【図 4 7】



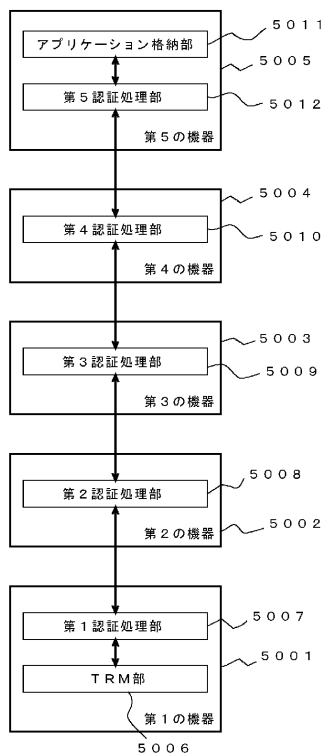
【図 48】



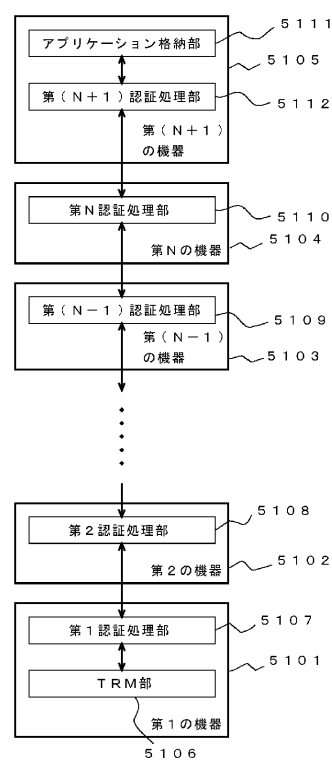
【図 49】



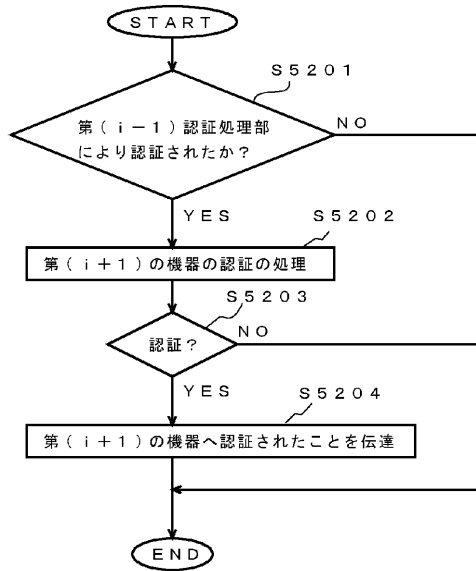
【図 50】



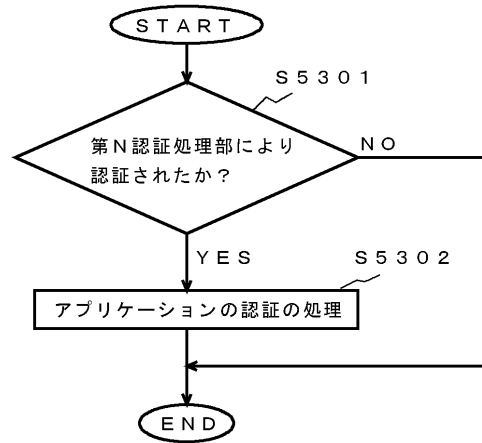
【図 51】



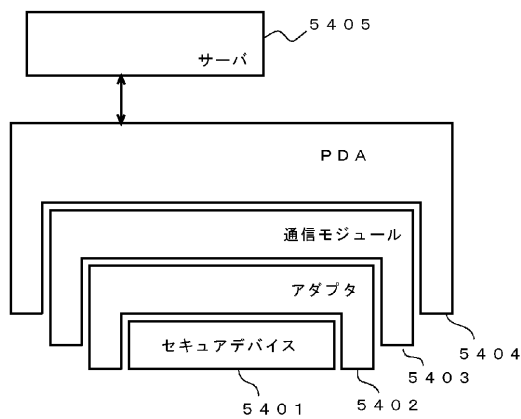
【図 5 2】



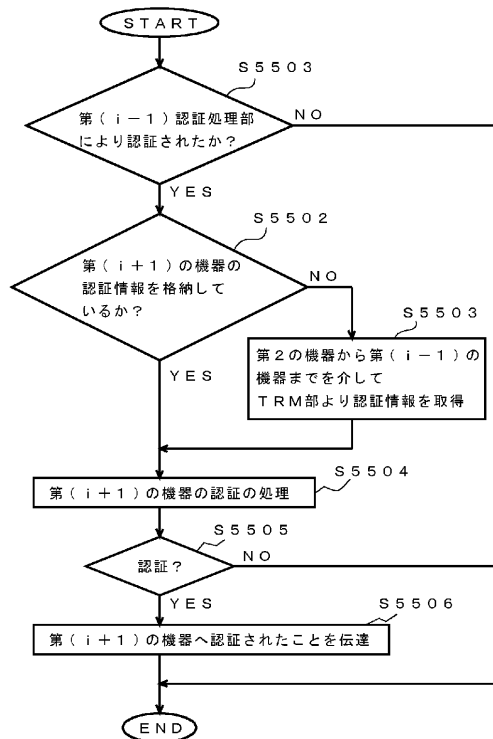
【図 5 3】



【図 5 4】



【図 5 5】



フロントページの続き

- (56)参考文献 特開平 1 0 - 3 3 3 9 0 2 (J P , A)
特開 2 0 0 1 - 3 1 2 4 0 2 (J P , A)
特開 2 0 0 0 - 2 5 0 8 6 0 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

G06F 21/22
G06K 17/00
G06K 19/00
H04L 9/10