(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2014-232359
(P2014-232359A)
(43)公開日　平成26年12月11日(2014.12.11)

(51)Int.Cl.		F I			テーマコード（参考）
G06F 21/31	(2013.01)	G06F	21/20	131A	5B084
G06F 21/62	(2013.01)	G06F	21/24	166A	
G06F 13/00	(2006.01)	G06F	13/00	510A	

審査請求 未請求　請求項の数 11　ＯＬ　（全 15 頁）

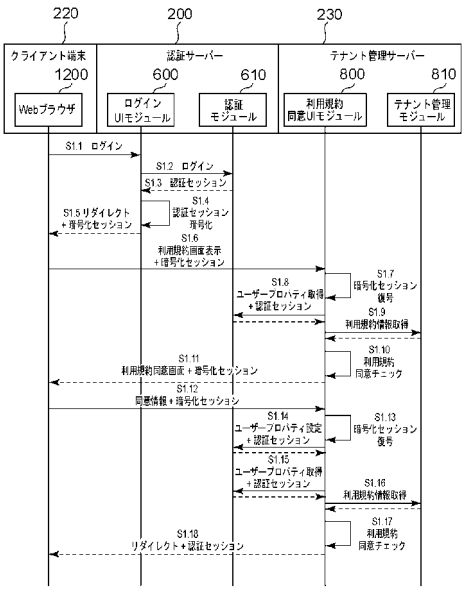
(21)出願番号　特願2013-111839 (P2013-111839)
(22)出願日　平成25年5月28日 (2013.5.28)

(71)出願人　000001007
キヤノン株式会社
東京都大田区下丸子３丁目３０番２号
(74)代理人　100126240
弁理士　阿部　琢磨
(74)代理人　100124442
弁理士　黒岩　創吾
(72)発明者　三原　誠
東京都大田区下丸子３丁目３０番２号キヤノン株式会社内

Ｆターム(参考)　5B084 AA01 AA12 AA30 AB30 AB34 AB36 BB16 CB04 CB22 CD24 CF12

(54)【発明の名称】情報処理サーバーシステム、制御方法、およびプログラム

(57)【要約】
【課題】　Ｗｅｂブラウザに認証セッションのｃｏｏｋｉｅを渡してしまうと、利用規約に同意していないのにも関わらずウェブサービスの利用が可能となってしまう。
【解決手段】　クライアントがウェブサービスを利用するために用いられる第１の認証セッションとは異なる第２の認証セッションを用いてユーザーが利用規約に同意したことを確認する情報処理サーバーシステムを提供する。
【選択図】　図６



【特許請求の範囲】
【請求項１】
ユーザーが認証されたことに応じて生成された、クライアントがウェブサービスを利用するために用いられる第１の認証セッションを基に、第２の認証セッションを生成する生成手段と、
生成された前記第２の認証セッションを前記クライアントへ送信する送信手段と、
前記ウェブサービスの利用規約に同意したことを示す情報とともに、前記送信手段により送信された前記第２の認証セッションを前記クライアントから受信する受信手段と、を有し、
前記送信手段は、受信された前記情報と前記第２の認証セッションとから前記ユーザーは前記ウェブサービスの利用規約に同意したことが確認されたことに応じて、前記第２の認証セッションに対応する前記第１の認証セッションを前記クライアントへ送信することを特徴とする情報処理サーバーシステム。
【請求項２】
前記情報処理サーバーシステムとは異なる情報処理サーバーシステムにおいてユーザーが認証されたことを示すレスポンスを認証サーバーが受信し、前記レスポンスを受信したことに応じて前記認証サーバーが前記クライアントへ送信する前記第１の認証セッションをフックするフック手段を更に有し、
前記生成手段は、前記フック手段によりフックされた前記第１の認証セッションを基に、前記第２の認証セッションを生成することを特徴とする請求項１に記載の情報処理サーバーシステム。
【請求項３】
前記ウェブサービスの利用規約に同意するための画面を提供する提供手段を更に有し、
前記提供手段は、認証されたユーザーが利用可能な複数の前記ウェブサービスであって、それらの利用規約に同意するための複数の画面を提供し、
前記送信手段は、認証されたユーザーが利用可能な複数の前記ウェブサービスの全ての利用規約に同意したことが確認されたことに応じて、前記第２の認証セッションに対応する前記第１の認証セッションを前記クライアントへ送信することを特徴とする請求項１または２に記載の情報処理サーバーシステム。
【請求項４】
前記提供手段は、認証された前記ユーザーが利用可能な前記ウェブサービスが存在しない場合、前記ユーザーのテナントに紐付く前記ウェブサービスの利用規約に同意するための画面を提供することを特徴とする請求項３に記載の情報処理サーバーシステム。
【請求項５】
前記生成手段は、前記第１の認証セッションを暗号化することで前記第２の認証セッションを生成し、
前記送信手段は、前記受信手段により受信された前記第２の認証セッションを復号化し、復号化されたことで得られる前記第１の認証セッションを前記クライアントへ送信することを特徴とする請求項１乃至４の何れか１項に記載の情報処理サーバーシステム。
【請求項６】
情報処理サーバーシステムを制御するための制御方法であって、
生成手段は、ユーザーが認証されたことに応じて生成された、クライアントがウェブサービスを利用するために用いられる第１の認証セッションを基に、第２の認証セッションを生成し、
送信手段は、生成された前記第２の認証セッションを前記クライアントへ送信する送信し、
受信手段は、前記ウェブサービスの利用規約に同意したことを示す情報とともに、前記送信手段により送信された前記第２の認証セッションを前記クライアントから受信し、
前記送信手段は、受信された前記情報と前記第２の認証セッションとから前記ユーザーは前記ウェブサービスの利用規約に同意したことが確認されたことに応じて、前記第２の

認証セッションに対応する前記第１の認証セッションを前記クライアントへ送信することを特徴とする制御方法。
【請求項７】
フック手段は、前記情報処理サーバーシステムとは異なる情報処理サーバーシステムにおいてユーザーが認証されたことを示すレスポンスを認証サーバーが受信し、前記レスポンスを受信したことに応じて前記認証サーバーが前記クライアントへ送信する前記第１の認証セッションをフックし、
前記生成手段は、前記フック手段によりフックされた前記第１の認証セッションを基に、前記第２の認証セッションを生成することを特徴とする請求項６に記載の制御方法。
【請求項８】
提供手段は、前記ウェブサービスの利用規約に同意するための画面を提供し、
前記提供手段は、認証されたユーザーが利用可能な複数の前記ウェブサービスであって、それらの利用規約に同意するための複数の画面を提供し、
前記送信手段は、認証されたユーザーが利用可能な複数の前記ウェブサービスの全ての利用規約に同意したことが確認されたことに応じて、前記第２の認証セッションに対応する前記第１の認証セッションを前記クライアントへ送信することを特徴とする請求項６または７に記載の制御方法。
【請求項９】
前記提供手段は、認証された前記ユーザーが利用可能な前記ウェブサービスが存在しない場合、前記ユーザーのテナントに紐付く前記ウェブサービスの利用規約に同意するための画面を提供することを特徴とする請求項８に記載の制御方法。
【請求項１０】
前記生成手段は、前記第１の認証セッションを暗号化することで前記第２の認証セッションを生成し、
前記送信手段は、前記受信手段により受信された前記第２の認証セッションを復号化し、復号化されたことで得られる前記第１の認証セッションを前記クライアントへ送信することを特徴とする請求項６乃至９の何れか１項に記載の制御方法。
【請求項１１】
請求項６乃至１０の何れか１項に記載の制御方法をコンピュータに実行させるためのプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
ウェブサービスの利用規約への同意に従いウェブサービスの利用を開始する情報処理サーバーシステム、制御方法、およびプログラム
【背景技術】
【０００２】
近年、クラウド型サービスを始めとする、インターネット上に設置されたサーバーを利用し顧客にサービスを提供するビジネスが多く存在する。このようなビジネスでは複数の異なるサービスが提供され、顧客はそれらサービスから自身が利用したいものを選択し、必要なサービスとだけ契約するといった契約形態が取られる。
【０００３】
また、このようなサービスでは、ある顧客企業にサービスを提供する場合、サービス提供側はテナントを新規に作成しその顧客企業に割り当てる。また新規作成したテナントを顧客企業側で管理するための初期ユーザーを作成し、テナントに登録する。顧客企業側の管理者は、作成された初期ユーザーとしてサービスにログインし、割り当てられたテナントにユーザーを追加するほか、必要な設定を行うことで、顧客企業がサービスの利用を開始できる。
【０００４】
サービスを実際に利用するユーザーはサービスに初回ログインする際に、サービス提供

者が定める利用規約や個人情報の同意が求められ、それに同意して、初めてサービスにログインし、サービスの利用が可能となる場合がある。これら利用規約は、各サービスで異なる利用規約をそれぞれ同意する場合や、サービス共通の一つの利用規約に同意すれば全てのサービスが利用可能となるケースが考えられる。

【０００５】

一般的に認証機能で保護されたサーバーへのアクセスは、サービスにログインした結果認証が成功したことを示す認証セッションをｃｏｏｋｉｅとしてクライアントのＷｅｂブラウザに保存させ、そのｃｏｏｋｉｅを持って行われる。サーバーが提供する各Ｗｅｂページへアクセスする際は、クライアントからサーバーへｃｏｏｋｉｅが送信されることで、サーバーは一連のＷｅｂページへのアクセスが同一ユーザーからのものであると特定しサービスを提供することが可能となる。特許文献１にて述べられているように、認証セッションのｃｏｏｋｉｅがクライアントのＷｅｂブラウザへ渡されると、そのＷｅｂブラウザは認証機能により保護されているＷｅｂページへのアクセスが可能となる。

【先行技術文献】

【特許文献】

【０００６】

【特許文献１】特許第４０５６３９０号

【発明の概要】

【発明が解決しようとする課題】

【０００７】

利用するユーザーに応じて利用規約や個人情報の同意を求める場合、ユーザーがサーバーのログイン画面にてログインを行う。サーバーではそのユーザーがどのサービスを利用可能なのか、およびどのサービスの利用規約に同意しているのかの情報を取得し、同意していないと判断した利用規約の同意画面をユーザーに提供する。同意画面を介してユーザーの同意結果がサーバーへ送信された際に、サーバーは利用規約に同意したユーザーを特定するために認証セッションのｃｏｏｋｉｅが必要となる。

【０００８】

しかし、Ｗｅｂブラウザに認証セッションのｃｏｏｋｉｅを渡してしまうと、利用規約に同意していないのにも関わらずウェブサービスの利用が可能となってしまう可能性がある。具体的には、同意画面表示中にユーザーがＷｅｂブラウザにてサービスのＵＲＬを直接指定すると、利用規約に同意することなくウェブサービスへアクセスし利用が可能となる。

【０００９】

本発明では上述の課題を鑑み、クライアントがウェブサービスを利用するために用いられる認証セッションとは異なる認証セッションを用いてユーザーが利用規約に同意したことを確認する情報処理サーバーシステムを提供する。

【課題を解決するための手段】

【００１０】

本発明の一実施形に係る情報処理サーバーシステムは、ユーザーが認証されたことに応じて生成された、クライアントがウェブサービスを利用するために用いられる第１の認証セッションを基に、第２の認証セッションを生成する生成手段と、生成された前記第２の認証セッションを前記クライアントへ送信する送信手段と、前記ウェブサービスの利用規約に同意したことを示す情報とともに、前記送信手段により送信された前記第２の認証セッションを前記クライアントから受信する受信手段と、を有し、前記送信手段は、受信された前記情報と前記第２の認証セッションとから前記ユーザーは前記ウェブサービスの利用規約に同意したことが確認されたことに応じて、前記第２の認証セッションに対応する前記第１の認証セッションを前記クライアントへ送信することを特徴とする。

【発明の効果】

【００１１】

クライアントがウェブサービスを利用するために用いられる認証セッションとは異なる

認証セッションを用いてユーザーが利用規約に同意したことを確認する情報処理サーバーシステムを提供できる。

【図面の簡単な説明】

【００１２】

【図１】システム構成図。

【図２】各装置のハードウェア構成図。

【図３】各装置のソフトウェアモジュール構成図。

【図４】認証サーバーで管理するテーブル構造

【図５】テナント管理サーバーで管理するテーブル構造

【図６】ログインおよび利用規約同意シーケンス図

【図７】利用規約の同意が必要かを確認するフローチャート

【図８】利用規約に関する画面

【図９】Ｓｉｎｇｌｅ　Ｓｉｇｎ　Ｏｎ　および利用規約同意画面表示シーケンス図

【図１０】ＳＡＭＬ検証成功レスポンスの判別処理のフローチャート

【図１１】認証サーバーで管理するテンポラリセッション管理テーブル構造

【発明を実施するための形態】

【００１３】

以下、本発明を実施するための最良の形態について図面を用いて説明する。

【００１４】

本実施の形態においては、インターネット上で帳票を生成する帳票サービス、生成した帳票を画像形成装置にて印刷するための印刷サービスが、インターネット上のサーバーに設置されていることを想定している。以降、これらのサービスのように、インターネット上で機能を提供しているサービスを、ウェブサービスと呼ぶ。

【実施例１】

【００１５】

実施例１における利用規約管理システムは、図１に示すような構成のネットワーク上に実現される。１００は、Ｗｉｄｅ　Ａｒｅａ　Ｎｅｔｗｏｒｋ（ＷＡＮ１００）であり、本発明ではＷｏｒｌｄ　Ｗｉｄｅ　Ｗｅｂ（ＷＷＷ）システムが構築されている。１０１は各構成要素を接続するＬｏｃａｌ　Ａｒｅａ　Ｎｅｔｗｏｒｋ（ＬＡＮ１０１）である。

【００１６】

２００はユーザーを認証する認証サーバーである。２１０はリソースサーバーであり、帳票サービスや印刷サービスと言ったウェブサービスが設置されている。なお１台のリソースサーバーに設置されるウェブサービスは１つでもよく、複数でもよい。また、実施例１において各サーバーは１台ずつ設置されているが複数台で構成されていても良く、そのため情報処理サーバーシステムと称した場合は少なくとも１台のサーバーを指していることになる。２２０はクライアント端末であり、Ｗｅｂブラウザがインストールされている。２３０はテナント管理サーバーであり、利用規約のコンテンツ管理、同意画面生成を行う。２４０はシングルサインオンのＳＭＡＬにおけるＩｄｅｎｔｉｔｙ　Ｐｒｏｖｉｄｅｒ（ＩｄＰ）であり、本システムとは別に提供される認証サーバーである。また、認可サーバー２００、リソースサーバー２１０、クライアント端末２２０、テナント管理サーバー２３０、ＩｄＰ２４０はそれぞれＷＡＮ１００およびＬＡＮ１０１を介して接続されている。なお認可サーバー２００、リソースサーバー２１０、クライアント端末２２０、テナント管理サーバー２３０、ＩｄＰ２４０はそれぞれ個別のＬＡＮ上に構成されていてもよいし同一のＬＡＮ上に構成されていてもよい。また認可サーバー２００、リソースサーバー２１０、テナント管理サーバー２３０は同一のサーバー上に構成されていてもよい。

【００１７】

なお、上述の情報処理サーバーシステムとは、ユーザー認証処理を行う少なくとも１台のログイン制御サーバーと、ログイン制御サーバーによるユーザー認証処理が成功したことに応じてサービスを提供するリソースサーバーとを含むシステムを指す。しかしながら

、それらのサーバーを１台に集約した形態も想定されるため、情報処理サーバーシステムと称する場合、複数のサービスを提供する形態が必ずしも複数台のサーバーから構成されるとは限らない。また、情報処理サーバーシステムは、ログイン制御サーバーのみ、またはリソースサーバーのみから構成されていても良い。

【００１８】

図２は本実施の形態に係るクライアント端末２２０の構成を示す図である。また認証サーバー２００、リソースサーバー２１０、テナント管理サーバー２３０、ＩｄＰ２４０のサーバーコンピューターの構成も同様である。尚、図２に示されるハードウェアブロック図は一般的な情報処理装置のハードウェアブロック図に相当するものとし、本実施形態のクライアント端末２２０およびサーバーコンピューターには一般的な情報処理装置のハードウェア構成を適用できる。

【００１９】

図２において、ＣＰＵ２３１は、ＲＯＭ２３３のプログラム用ＲＯＭに記憶された、或いはハードディスク（ＨＤ）等の外部メモリ２４１からＲＡＭ２３２にロードされたＯＳやアプリケーション等のプログラムを実行する。またＣＰＵ２３１は、システムバス２３４に接続される各ブロックを制御する。ここでＯＳとはコンピュータ上で稼動するオペレーティングシステムの略語であり、以下オペレーティングシステムのことをＯＳと呼ぶ。後述する各シーケンスの処理はこのプログラムの実行により実現できる。ＲＡＭ２３２は、ＣＰＵ２３１の主メモリ、ワークエリア等として機能する。キーボードコントローラ（ＫＢＣ）２３５は、キーボード２３９や不図示のポインティングデバイスからのキー入力を制御する。ＣＲＴコントローラ（ＣＲＴＣ）２３６は、ＣＲＴディスプレイ２４０の表示を制御する。ディスクコントローラ（ＤＫＣ）２３７は各種データを記憶するハードディスク（ＨＤ）等の外部メモリ２４１におけるデータアクセスを制御する。ネットワークコントローラ（ＮＣ）２３８はＷＡＮ１００もしくはＬＡＮ１０１を介して接続されたサーバーコンピューターや他の機器との通信制御処理を実行する。尚、後述の全ての説明においては、特に断りのない限りサーバーにおける実行のハード上の主体はＣＰＵ２３１であり、ソフトウェア上の主体は外部メモリ２４１にインストールされたアプリケーションプログラムである。

【００２０】

図３は実施例１に係る、認証サーバー２００、リソースサーバー２１０、クライアント端末２２０、テナント管理サーバー２３０、ＩｄＰ２４０、それぞれのモジュール構成を示す図である。認証サーバー２００はログインＵＩモジュール６００と認証モジュール６１０、ＳＳＯフックモジュールを持つ。リソースサーバー２１０はリソースサーバーモジュール７００を持つ。クライアント端末２２０はＷＷＷを利用するためのユーザーエージェントであるＷｅｂブラウザ１２００を持つ。テナント管理サーバー２３０は利用規約ＵＩモジュール８００、テナント管理モジュール８１０を持つ。ＩｄＰ２４０はログインＵＩモジュール９００と認証モジュール９１０を持つ。

【００２１】

図４は認証サーバー２００が外部メモリに記憶するデータテーブルである。これらデータテーブルは認証サーバー２００の外部メモリではなく、ＬＡＮ１０１を介して通信可能に構成された別のサーバーに記憶するよう構成する事も出来る。ユーザー管理テーブル１２００は、ユーザーＩＤ１２０１、パスワード１２０２、テナントＩＤ１２０３、ロール１２０４、利用規約同意情報１２０５、セッション情報１２０６から成る。認証サーバー２００は、ユーザーＩＤ１２０１、パスワード１２０２の情報の組を検証し各ユーザーを認証し認証セッションを生成する機能を備える。クライアント端末２２０は、認証セッションを利用することでウェブサービスへのアクセスが可能となる。ロール１２０４はそれぞれのユーザーがどういった権限を持つかを示す情報である。”ＣｕｓｔｏｍｅｒＡｄｍｉｎ”は管理者の権限、”Ｃｕｓｔｏｍｅｒ”は一般者の権限、”Ｆｏｒｍ”は帳票サービスを利用するための権限、”Ｐｒｉｎｔ”は印刷サービスを利用するための権限である。”Ｆｏｒｍ”や”Ｐｒｉｎｔ”のロールがあることで初めて対応するウェブサービスが

利用可能である。利用規約同意情報１２０５はそれぞれのユーザーがどの利用規約に同意したかを示す情報である。セッション情報１２０６は、生成した認証セッションを格納する領域で、システム一意に決まる認証セッションのＩＤや認証セッションの有効期限が格納される。

【００２２】

図５ａ、図５ｂはテナント管理サーバー２３０が外部メモリに記憶するデータテーブルである。これらデータテーブルは、テナント管理サーバー２３０の外部メモリではなく、ＬＡＮ１０１を介して通信可能に構成された別のサーバーに記憶するよう構成する事も出来る。図５ａはライセンス管理テーブル１５００である。ライセンス管理テーブル１５００はテナントＩＤ１５０１、販売テナントＩＤ１５０２、ライセンス１５０３、ライセンス数１５０４から成る。ライセンス管理テーブル１５００では、顧客のテナントがどのウェブサービスを利用できるかを管理している。実施例１では、テナントＩＤ１５０１　“１００１ＡＡ”の顧客テナントが、販売テナントＩＤ１５０２　“１０１ＡＡ”の販売テナントから、”Ｆｏｒｍ”と”Ｐｒｉｎｔ”のライセンス１５０３をライセンス数１５０４　“２０”利用できるという情報が保持されている。

【００２３】

図５ｂは利用規約管理テーブル１６００である。利用規約管理テーブル１６００は、利用規約ＩＤ１６０１、販売テナントＩＤ１６０２、ライセンス１６０３、リビジョン１６０４、コンテンツ１６０５から成る。利用規約管理テーブル１６００では、ライセンスを販売する販売テナント毎に、ライセンスに対応した利用規約を管理している。利用規約ＩＤ１６０１は利用規約をシステム一意に識別するＩＤである。販売テナントＩＤ１６０２はどの販売テナントから販売された場合の設定かを管理する。ライセンス１６０３は、利用規約を表示すべきライセンスを管理する。本実施例では、”Ｆｏｒｍ”ライセンス用、”Ｐｒｉｎｔ”ライセンス用、”Ｆｏｒｍ”と”Ｐｒｉｎｔ”ライセンスで共用、といった利用規約が定義されているリビジョン１６０４では、各利用規約のリビジョンを管理している。リビジョンの情報は、ユーザーが同意済みの利用規約がリビジョンアップされた場合に、新しいリビジョンの利用規約に再度同意を求める処理を実現するために保持している。コンテンツ１６０５は、実際にユーザーに同意を求める利用規約の内容を管理している。

【００２４】

ユーザーがＷｅｂページよりログインを行い、利用規約に同意してウェブサービスを利用開始するまでの一連の方法に関する本実施形態のシーケンスを図６にて説明する。本シーケンスは、クライアント端末２２０のＷｅｂブラウザ１２００を利用して情報処理サーバーシステムにログインする際に実行される処理である。

【００２５】

まず、Ｗｅｂブラウザ１２００は認証サーバー２００のログインＵＩモジュール６００へアクセスしログインを行う（Ｓ１．１）。本処理では、システム利用者はユーザーＩＤおよびパスワードと言ったユーザー認証情報を入力する。ログインＵＩモジュール６００はユーザーＩＤとパスワードを認証モジュール６１０へ通知する。（Ｓ１．２）。認証モジュール６１０は受信したユーザーＩＤとパスワードの一致をユーザー管理テーブル１２００のデータで確認し認証が成功したら認証セッションを生成する。認証モジュール６１０は生成した認証セッションをユーザー管理テーブル１２００のセッション情報１２０６に格納した後、ログインＵＩモジュール６００にレスポンスする（Ｓ１．３）。ログインＵＩモジュール６００はステップＳ１．３にて取得した認証セッションを暗号化する（Ｓ１．４）。なお、本暗号化に利用する暗号鍵は、ログインＵＩモジュール６００と利用規約同意ＵＩモジュール８００のみで共有されている。よって、認証セッションへの暗号化と復号化はログインＵＩモジュール６００と利用規約同意ＵＩモジュール８００でのみ実施できる。ログインＵＩモジュール６００は暗号化セッションをｃｏｏｋｉｅに設定し、利用規約画面へのリダイレクトをクライアント端末２２０にレスポンスする（Ｓ１．５）。

【００２６】
Ｗｅｂブラウザ１２００はリダイレクトの指示を受け、テナント管理サーバー２３０の利用規約同意ＵＩモジュール８００に対して利用規約同意画面取得のリクエストを行う。その際、暗号化セッションの情報も合わせて送信する（Ｓ１．６）。利用規約同意ＵＩモジュール８００はＷｅｂブラウザ１２００のリクエストから暗号化セッションを取得し復号化処理を行い、認証セッションの情報を取得する（Ｓ１．７）。利用規約同意ＵＩモジュール８００は取得した認証セッションの情報を認証モジュール６１０に送信し、ユーザープロパティを取得する（Ｓ１．８）。認証モジュール６１０は、ユーザー管理テーブル１２００のセッション情報１２０６から該当の認証セッションを持つユーザーを特定し、ユーザーＩＤ１２０１、パスワード１２０２、テナントＩＤ１２０３、ロール１２０４、利用規約同意情報１２０５の各データを取得する。認証モジュール６１０は取得した情報を利用規約同意ＵＩモジュール８００へレスポンスする（Ｓ１．８）。利用規約同意ＵＩモジュール８００はＳ１．８で取得したテナントＩＤ１２０３の情報をテナント管理モジュール８１０に問い合わせ、利用規約情報を取得する（Ｓ１．９）。テナント管理モジュール８１０は、ライセンステーブル１５００および、利用規約管理テーブル１６００より、対象のテナントで同意が必要な利用規約の情報を取得する。例えば、テナントＩＤとして”１００１ＡＡ”が渡された場合、利用規約ＩＤ１６０１が”２”（１０１ＡＡが販売したＦｏｒｍライセンスの最新リビジョンの利用規約）と”３”（１０１ＡＡが販売したＰｒｉｎｔライセンスの最新リビジョン）の利用規約情報が取得される。テナント管理モジュール８１０は、取得した利用規約情報を利用規約同意ＵＩモジュール８００にレスポンスする（Ｓ１．９）。利用規約同意ＵＩモジュール８００は、Ｓ１．８で取得したユーザープロパティとＳ１．１０で取得した利用規約情報を利用し、同意が必要な利用規約が存在するかをチェックする（Ｓ１．１０）。

【００２７】
ここで、図７にてＳ１．１０の利用規約の存在チェックの処理の流れの詳細を示す。本処理では、管理者と一般者で異なる判定で利用規約の存在をチェックする。一般者は、ライセンス付与されたウェブサービスを利用するために必ず対応するロールが割り当てられるので、そのロールを元に判定する。管理者はユーザー管理やテナント管理といった、特定のウェブサービスのロールを持たない場合がある。なぜなら、管理者はウェブサービスの利用を前提としたアカウントではなく、実際にウェブサービスを利用する同じテナント内のユーザーを管理するためのアカウントであるからである。そのため、ライセンスに対応するロールを持たない場合でも利用規約に同意させシステムにログインさせる必要がある。よって、管理者はロールではなく、管理者が所属するテナントに販売されたライセンスの有無を元に利用規約の判断を行う。

【００２８】
Ｓ１．１０ではユーザープロパティより、ユーザーが管理者か一般者かを判断する（Ｓ２．１）。ユーザーが一般者の場合にはＳ２．２に進み、ユーザーに割り当てられたロールを元に利用規約の判定を行う。以降は、ユーザー管理テーブル１２００で定義されたユーザーの情報を元に説明する。Ｓ２．２ではユーザーにライセンスに対応したロールが割り当てられているかをチェックする。もしロールが割り当てられていなければＳ２．５に進みそのユーザーのログインを許さずシステム利用を禁止する。”Ｕｓｅｒ２”の場合では”Ｐｒｉｎｔ”のロールが割り当てられているのでＳ２．３に進む。Ｓ２．３では、ユーザーに割り当てられたロールの数だけループ処理を行う。”Ｕｓｅｒ２”の場合は”Ｐｒｉｎｔ”分の１回だけループし、”Ｕｓｅｒ３”の場合は、”Ｆｏｒｍ”と”Ｐｒｉｎｔ”分の２回ループする。Ｓ２．４では、対応する利用規約が同意済みか否かをチェックする。”Ｕｓｅｒ２”の場合、テナントＩＤ “１００１ＡＡ”に所属しているので、ライセンステーブル１５００の情報より販売テナントＩＤ “１０１ＡＡ”よりテナントに紐付くウェブサービスであって、対象のテナントに販売されている”Ｐｒｉｎｔ”のライセンスを特定する。さらに利用規約管理テーブル１６００の情報より利用規約ＩＤ１６０１ “３”の利用規約を特定する。最後に”Ｕｓｅｒ２”の利用規約同意情報１２０５に

該当の利用規約に同意した情報が記録されていないので、Ｓ２．７の同意が必要な利用規約が存在する処理に進む。”Ｕｓｅｒ１”のように対応する利用規約が同意済みの場合にはＳ２．６の同意が必要な利用規約が存在しない処理に進む。本処理まででユーザーが一般者の場合に利用規約への同意処理が必要か否かの判断処理が完了する。

【００２９】

Ｓ２．１に戻る。ユーザーが管理者の場合にはＳ２．１０に進み、ユーザーが所属するテナントに販売されたライセンスを元に利用規約の判定を行う。ウェブサービスＳ２．１０では、ユーザーが所属するテナントに割り当てられたライセンス分ループ処理を行う。”Ａｄｍｉｎ１”の場合は、テナントＩＤ　”１００１ＡＡ”テナントに所属するので、ライセンステーブル１５００の情報より”Ｆｏｒｍ”と”Ｐｒｉｎｔ”のライセンスの種類分の２回ループする。本処理により、ロールの割り当てられていない管理者であっても適切な利用規約を取得できる。Ｓ２．１１では、対応する利用規約が同意済みかをチェックする。”Ａｄｍｉｎ１”の場合、ライセンステーブル１５００の情報より販売テナントＩＤ　“１０１ＡＡ”を特定する。さらに利用規約管理テーブル１６００の情報より利用規約ＩＤ　２，３の利用規約を特定する。最後に”Ａｄｍｉｎ１”の利用規約同意情報１２０５に該当の利用規約に同意した情報が記録されているかをチェックする。本例ではすでに同意済みなので、Ｓ２．１３の同意が必要な利用規約が存在しない処理に進む。もし、利用規約が同意済みではない場合にはＳ２．１２の同意が必要な利用規約が存在する処理に進む。本処理まででユーザーが管理者の場合に利用規約への同意処理が必要か否かの判断処理が完了する。以上がＳ１．１０で行われる同意が必要な利用規約の存在をチェックするための詳細な処理の流れである。

【００３０】

図６のＳ１．１０以降の処理の説明に戻る。Ｓ１．１０にて同意が必要な利用規約が存在した場合、利用規約同意ＵＩモジュール８００は、コンテンツ１６０５のデータより利用規約同意画面を生成し、Ｓ１．４にて生成した暗号化セッションをｃｏｏｋｉｅに設定し、クライアント端末２２０にレスポンスする。図８ａ、図８ｂの８０００と８０１０が利用規約同意画面の実施形態の例である。図８ａは利用規約への同意のみ求める場合の画面の例である。利用規約に同意しなければシステムの利用ができないので、本画面の様に同意するボタンのみの画面提供のみでも十分である。もし、利用規約に同意したくない場合には、Ｗｅｂブラウザ１２００を終了する等で処理を終了することになる。図８ｂは利用規約への同意と不同意を求める場合の画面である。利用規約に不同意の場合に何らかの処理（例えばメッセージを表示する等）を実施したい場合にはこちらの画面を利用する。どちらも同意した場合の処理に差異はない。

【００３１】

８００１、８０１１にコンテンツ１６０５のデータが表示され、８００２、８０１２に同意ボタンが用意される。また、８０１３に同意しないボタンが用意される。利用規約同意画面８０００、８０１０の同意ボタン８００２、８０１２もしくは、同意しないボタン８０１３が押下されたら、Ｗｅｂブラウザ１２００はテナント管理サーバー２３０の利用規約同意ＵＩモジュール８００に対して同意情報の通知リクエストを行う。その際、暗号化セッションの情報も合わせて送信する（Ｓ１．１２）。利用規約同意ＵＩモジュール８００はＷｅｂブラウザ１２００のリクエストから同意情報を取得する。同意されていなかった場合には、暗号化セッションを削除しエラー画面をクライアントにレスポンスする。同意されていた場合にはリクエストから暗号化セッションを取得し復号化処理を行い、認証セッションの情報を取得する（Ｓ１．１３）。利用規約同意ＵＩモジュール８００は取得した認証セッションと同意した利用規約のＩＤを認証モジュール６１０に送信し、ユーザープロパティを設定する（Ｓ１．１４）。

【００３２】

認証モジュール６１０は、ユーザー管理テーブル１２００のセッション情報１２０６から該当の認証セッションを持つユーザーを特定し、利用規約同意情報１２０５に利用規約のＩＤを設定する。利用規約同意ＵＩモジュール８００はさらにＳ１．１５、Ｓ１．１６

、Ｓ１．１７にてさらに同意すべき利用規約が存在するかをチェックする。本チェックはＳ１．８、Ｓ１．９、Ｓ１．１０と同様の処理である。利用規約同意ＵＩモジュール８００は、Ｓ１．１７にて同意が必要な利用規約が存在しなかった場合、ｃｏｏｋｉｅに認証セッションを設定し、リソースサーバー２１０で提供されるウェブサービスに対するリダイレクトをクライアント端末２２０にレスポンスする（Ｓ１．１８）。クライアント端末２２０はユーザーがすべての利用規約に同意した後、初めて、認証セッションをサーバーから取得することが可能となる。これにより、認証セッションを必要とする各ウェブサービスへのアクセスが可能となり、クライアント端末２２０は情報処理サーバーシステム内のウェブサービスの利用を開始できる。

【００３３】

以上が、ユーザーがＷｅｂページよりログインを行い、利用規約に同意してウェブサービスを利用開始するまでの一連の方法に関する本実施形態のシーケンスの説明である。

【実施例２】

【００３４】

実施例２として、本情報処理サーバーシステムがＳｅｒｖｉｃｅ　Ｐｒｏｖｉｄｅｒ（ＳＰ）となり、別の情報処理サーバーシステムのＩｄｅｎｔｉｔｙ　Ｐｒｏｖｉｄｅｒ（ＩｄＰ）と　ＳＡＭＬ（Ｓｅｃｕｒｉｔｙ　Ａｓｓｅｒｔｉｏｎ　Ｍａｒｋｕｐ　Ｌａｎｇｕａｇｅ）によるＳｉｎｇｌｅ　Ｓｉｇｎ　Ｏｎ（ＳＳＯ）を実現している環境での利用規約同意方法に関して説明する。前提として、認証サーバー２００とＩｄＰ２４０は事前にＳＡＭＬによるＳＳＯに必要な設定が全てなされている。また、ＳＳＯフックモジュール６２０は認証サーバーのＷｅｂページへのアクセスのレスポンスを全てフックするように設定されている。本フックの設定は、認証サーバー２００のＨＴＴＰ機能をつかさどるＷｅｂサーバーに対して行う。一般的なＷｅｂサーバーは外部モジュールを追加することで、ＨＴＴＰ機能の処理途中に自由に処理を追加することが可能である。ＳＳＯフックモジュール６２０は外部モジュールとして作成されており、Ｗｅｂサーバーの、全てのＨＴＴＰレスポンスをクライアント端末２２０に返すタイミングの処理に組み込まれている。

【００３５】

ユーザーがＩｄＰのＷｅｂページよりログインを行い、クライアント端末２２０がＳＡＭＬによるＳＳＯで本情報処理サーバーシステムにアクセスし利用規約同意画面を表示するまでの一連の処理方法について図９を用いて説明する。まず、ＩｄＰ２４０のログインＵＩモジュール９００へアクセスしログインを行う（Ｓ３．１）。ログインＵＩモジュール９００はログイン処理を行い、ＳＡＭＬレスポンスの生成を行う。一般的なＩＤＰで生成するＳＡＭＬレスポンスでは、認証したユーザーを識別する情報等が含まれており、さらにはそのレスポンスは電子署名されている。ログインＵＩモジュール９００はＳＡＭＬレスポンスを本システムに対するリダイレクトの指示とともに、クライアント端末２２０へレスポンスを行う。クライアント端末２２０のＷｅｂブラウザ１２００はＳＡＭＬレスポンスとともに、認証サーバー２３０の認証モジュール６１０へＳＡＭＬ検証要求を行う。認証モジュール６１０は受け取ったＳＡＭＬレスポンスが正しいかを検証する。本検証ではＳＡＭＬレスポンスの電子署名が、事前に設定したＩＤＰで行われたものかを検証したうえで、含まれるユーザーを識別する情報を取得する。さらには、事前に設定したＩＤＰのユーザーと本情報処理サーバーシステムのユーザーのマッピング情報を元に、ＳＡＭＬレスポンスから取得したユーザーＩＤを本情報処理サーバーシステムにおけるユーザーのユーザーＩＤに変換してログインを許可し認証セッションを生成する。認証モジュール６１０は生成した認証セッションをユーザー管理テーブル１２００のセッション情報１２０６に格納した後、クライアント端末２２０にレスポンスする（Ｓ３．４）。ここで、認証サーバー２３０のＳＳＯフックモジュール６２０が認証サーバーの全てのレスポンスをフックするため、Ｓ３．４のレスポンスをフックする。ＳＳＯフックモジュール６２０はフックしたレスポンスがＳＡＭＬ検証成功レスポンスか否かをチェックする（Ｓ３．５）。

【００３６】
図１０にてＳ３．５の詳細な処理の流れを説明する。Ｓ４．１では、ＳＡＭＬ検証要求のレスポンスか否かを判断する。前述のとおり、ＳＳＯフックモジュール６２０は認証サーバー２００の全てのレスポンスの処理で実行されるため、例えばログインへのレスポンス等もフックする。よって、全のレスポンスの中からＳＡＭＬ検証のレスポンスを特定する必要がある。ＳＳＯフックモジュール６２０はＳＡＭＬ検証のためのＵＲＬを保持する。そのＵＲＬを利用し、フックしたレスポンスがそのＵＲＬへのリクエストに対するものか否かで判断を行う。例えば、ＳＳＯフックモジュール６２０がＳＡＭＬ検証のＵＲＬとして”／ａｕｔｈ／Ｓａｍｌ／ＳＰ／ＳＳＯ／Ｐｏｓｔ”を保持していた場合、フックしたレスポンスが、そのＵＲＬへのリクエストに対するレスオンスであるかが判別されることになる。Ｓ４．１でＵＲＬがマッチしなければＳ４．４に進み、ＳＳＯフックモジュール６２０は何も行わない。Ｓ４．１でＵＲＬがマッチした場合はＳ４．２に進み、さらにレスポンスのｃｏｏｋｉｅに認証セッションが含まれるかをチェックする。ＳＡＭＬ検証に成功するとシステムにアクセスするための認証セッションがレスポンスのｃｏｏｋｉｅに設定されクライアント端末２２０にレスポンスされるため、認証セッションの有無でＳＡＭＬ検証の成否が決定できる。ＳＡＭＬ検証に失敗している場合ではｃｏｏｋｉｅには認証セッションが含まれないので、Ｓ４．４に進む。認証セッションが含まれる場合にはＳ４．３に進み、ＳＡＭＬ検証の成功レスポンスとして処理を行う。

【００３７】
ＳＳＯフックモジュール６２０はＳＡＭＬ検証の成功レスポンス（Ｓ４．３）の処理として、図９　Ｓ３．６の認証セッションの暗号化処理を行う。本暗号化で利用する暗号鍵はログインＵＩモジュール６００と利用規約同意ＵＩモジュール８００で利用されるものと同じものである。Ｓ３．６では、ＳＳＯフックモジュール６２０は、まず、ＳＡＭＬ検証成功レスポンスのｃｏｏｋｉｅより認証セッションの取得と削除を実施する。次に、取得した認証セッションを暗号化しレスポンスのｃｏｏｋｉｅに設定する。さらには、レスポンスに含まれるＳＡＭＬ検証の処理で設定された、ＳＡＭＬ検証成功後のウェブサービスへのリダイレクト先ＵＲＬを利用規約同意画面表示のＵＲＬに書き換える。Ｓ３．６の処理の後、ＳＳＯフックモジュール６２０はクライアント端末２２０にレスポンスを返す（Ｓ３．７）。Ｗｅｂブラウザ１２００はリダイレクトの指示を受け、テナント管理サーバー２３０の利用規約同意ＵＩモジュール８００に対して利用規約同意画面取得のリクエストを行う。その際、暗号化セッションの情報も合わせて送信する（Ｓ３．８）。

【００３８】
以上が、ユーザーがＩｄＰのＷｅｂページよりログインを行い、ＳＡＭＬによるＳＳＯで本システムにアクセスし利用規約同意画面を表示するまでの一連の処理に関するシーケンスの説明である。Ｓ３．８の処理以降は、図６のＳ１．７以降の処理と同様となり、ＳＡＭＬ　ＳＳＯでの連携時でも利用規約の同意後にウェブサービスの利用を開始させることが実現可能となる。結果、本来であれば、クライアント端末２２０はＳＡＭＬによりウェブサービスへリダイレクトをしてサービスを受けることになるが、利用規約同意画面表示のＵＲＬにアクセスした結果、ユーザーは利用規約に同意しない限りクライアント端末２２０を介してウェブサービスを利用できなくなり、ウェブサービスの適正な提供が可能となる。

【実施例３】
【００３９】
実施例３では、認証セッションを暗号化セッションに暗号化して利用する手段の別の形態を説明する。認証サーバー２２０にて認証セッションに関連づけられたテンポラリセッションを生成し保持する方法であり、認証セッションを暗号化せずとも利用規約同意を行うことが可能となる。

【００４０】
図１１は認証サーバー２２０が外部メモリに記憶するデータテーブルである。これらデータテーブルは認証サーバー２００の外部メモリではなく、ＬＡＮ１０１を介して通信可

能に構成された別のサーバーに記憶するよう構成する事も出来る。テンポラリセッション管理テーブル１３００は、テンポラリセッション１３０１、認証セッション１３０２から成る。テンポラリセッション１３０１はシステムで一意に識別されるテンポラリセッションのＩＤが格納される。

【００４１】

実施例３では、実施例１、および２におけるＳ１．４、Ｓ３．６の認証セッション暗号化処理の代わりに次の処理を実施する。まず、ログインＵＩモジュール６００、およびＳＳＯフックモジュール６３０はＳ１．４、Ｓ３．６を処理する際、認証セッションを認証サーバー２２０に対して通知し、テンポラリセッションの生成を依頼する。依頼を受けた認証サーバー２２０は、テンポラリセッションを生成し、認証セッションの情報と関連付けてテンポラリセッション管理テーブル１３００にデータを格納したのち、テンポラリセッションをレスポンスする。テンポラリセッションを受け取ったログインＵＩモジュール６００やＳＳＯフックモジュール６３０は以降暗号化セッションの代わりにテンポラリセッションを利用する。次に、実施例１、および２におけるＳ１．７、Ｓ１．１３の暗号化セッション復号化処理の代わりに次の処理を実施する。利用規約同意ＵＩモジュール８００は、Ｓ１．７とＳ１．１３を処理する際、認証サーバー２２０にテンポラリセッションを通知し、認証セッションの取得を依頼する。依頼を受けた認証サーバー２２０は、テンポラリセッション管理テーブル１３００より、受領したテンポラリセッションに対応した認証セッションを取得しレスポンスする。認証セッションを受け取った利用規約同意ＵＩモジュール８００は、以降、復号化した認証セッションの代わりにテンポラリセッションより取得した認証セッションを利用する。以上が、認証セッションを暗号化セッションに暗号化して利用する手段の別の形態の説明となる。

【００４２】

＜その他の実施形態＞

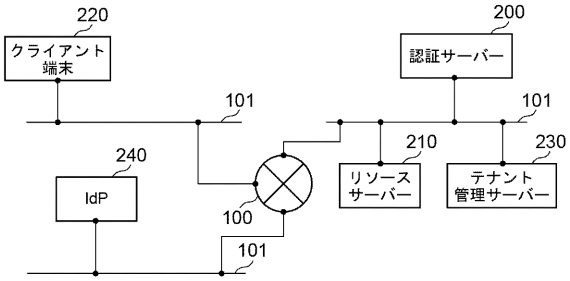
また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（またはＣＰＵやＭＰＵ等）がプログラムを読み出して実行する処理である。

【符号の説明】

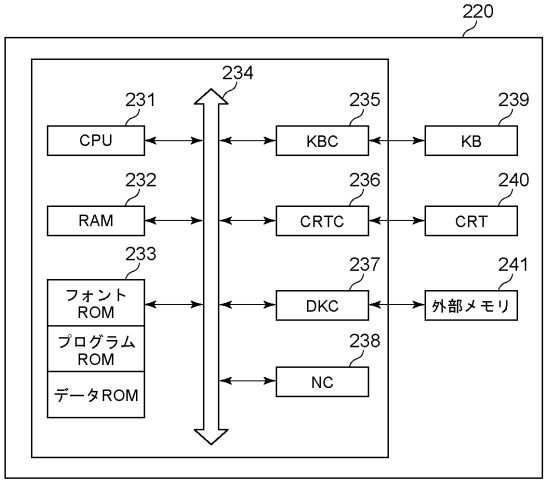
【００４３】

１００　ＷＡＮ
１０１　ＬＡＮ
２００　認証サーバー
２１０　リソースサーバー
２２０　クライアント端末
２３０　テナント管理サーバー
２４０　ＩｄＰ
６００　ログインＵＩモジュール
６１０　認証モジュール
６２０　ＳＳＯフックモジュール
７００　リソースサーバーモジュール
８００　利用規約同意ＵＩモジュール
８１０　テナント管理モジュール
９００　ログインＵＩモジュール
９１０　認証モジュール
１２００　Ｗｅｂブラウザ

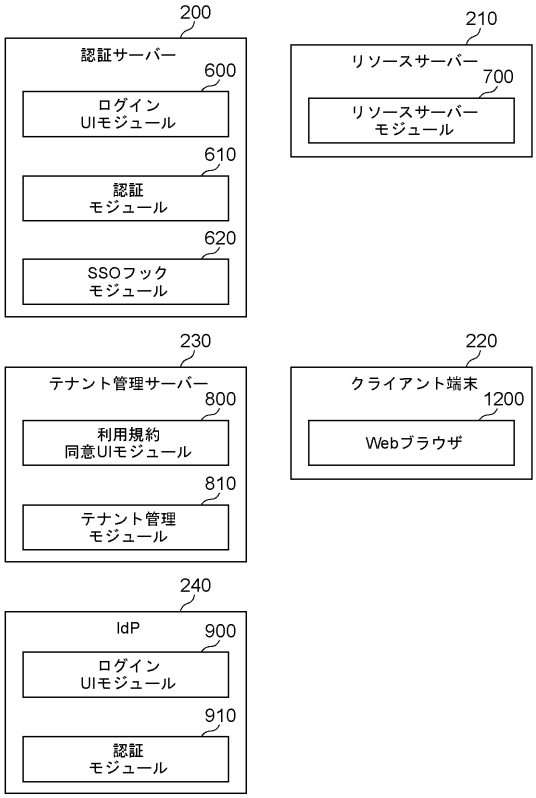
【図１】



【図２】



【図３】

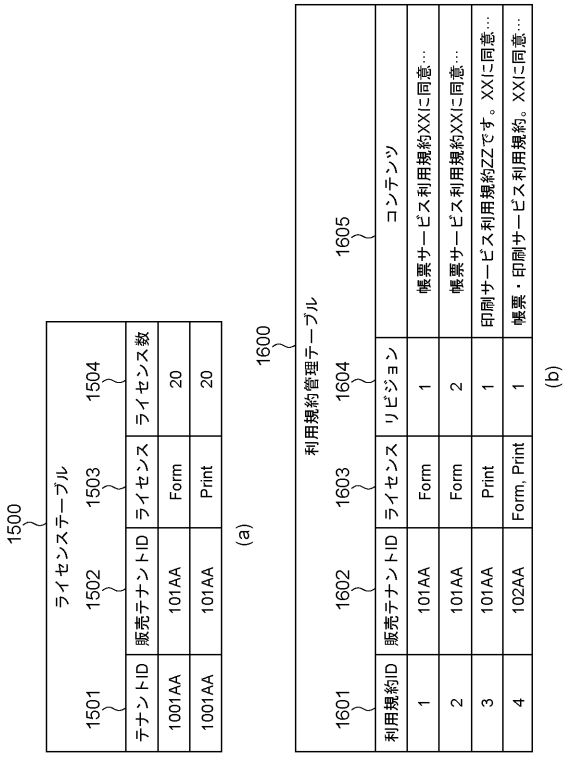


【図４】

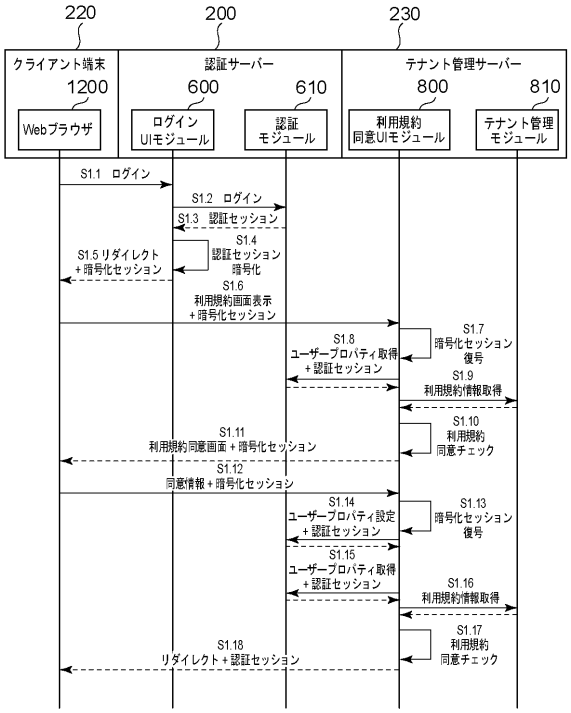
1200 ユーザー管理テーブル

1201 ユーザーID	1202 パスワード	1203 テナントID	1204 ロール	1205 利用規約同意情報	1206 セッション情報
Admin1	**********	1001AA	CustomerAdmin	2,3	XXXX,2013/04/16 07:07
User1	**********	1001AA	Customer, Form	2	YYYY,2013/04/16 07:07
User2	**********	1001AA	Customer, Print		
User3	**********	1001AA	Customer, Form, Print		

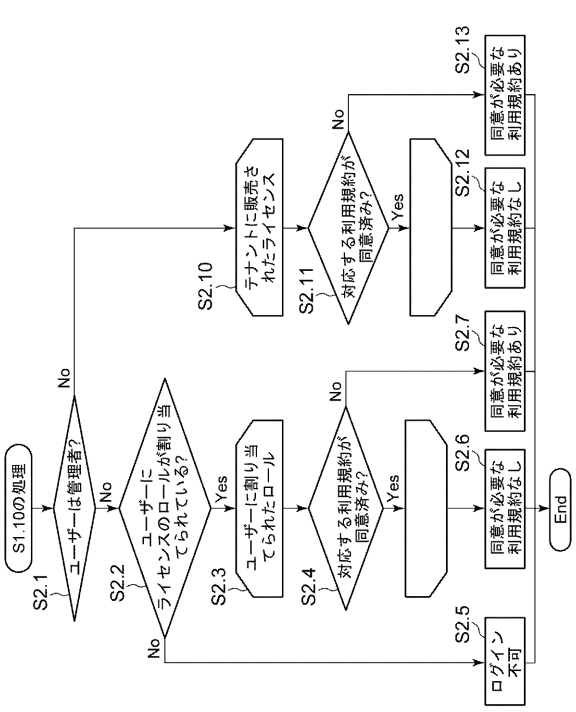
【図５】


1500
ライセンステーブル
1501 テナントID
1502 販売テナントID
1503 ライセンス
1504 ライセンス数
1001AA 101AA Form 20
1001AA 101AA Print 20
(a)
1600
利用規約管理テーブル
1601 利用規約ID
1602 販売テナントID
1603 ライセンス
1604 リビジョン
1605 コンテンツ
1 101AA Form 1 帳票サービス利用規約XXに同意…
2 101AA Form 2 帳票サービス利用規約XXに同意…
3 101AA Print 1 印刷サービス利用規約ZZです。XXに同意…
4 102AA Form, Print 1 帳票・印刷サービス利用規約。XXに同意…
(b)

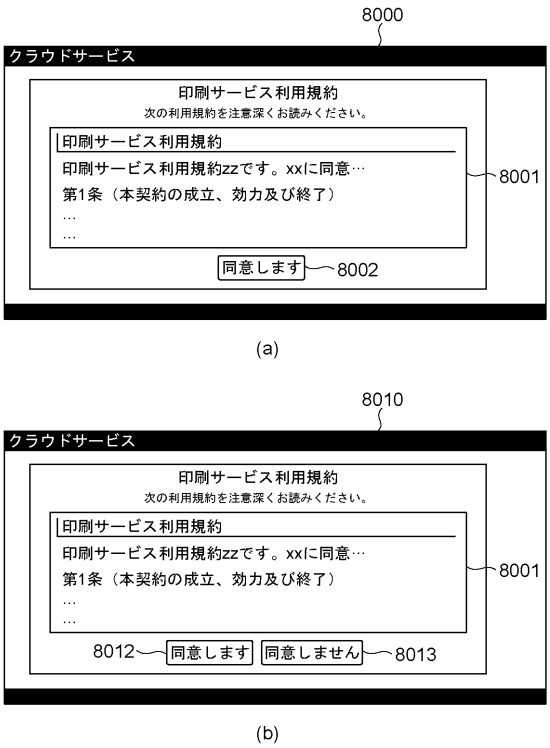
【図６】


220 200 230
クライアント端末 1200
認証サーバー 600 610
テナント管理サーバー 800 810
Webブラウザ
ログイン UIモジュール
認証 モジュール
利用規約 同意UIモジュール
テナント管理 モジュール
S1.1 ログイン
S1.2 ログイン
S1.3 認証セッション
S1.4 認証セッション 暗号化
S1.5 リダイレクト + 暗号化セッション
S1.6 利用規約画面表示 + 暗号化セッション
S1.7 暗号化セッション 復号
S1.8 ユーザープロパティ取得 + 認証セッション
S1.9 利用規約情報取得
S1.10 利用規約 同意チェック
S1.11 利用規約同意画面 + 暗号化セッション
S1.12 同意情報 + 暗号化セッション
S1.13 暗号化セッション 復号
S1.14 ユーザープロパティ設定 + 認証セッション
S1.15 ユーザープロパティ取得 + 認証セッション
S1.16 利用規約情報取得
S1.17 利用規約 同意チェック
S1.18 リダイレクト + 認証セッション

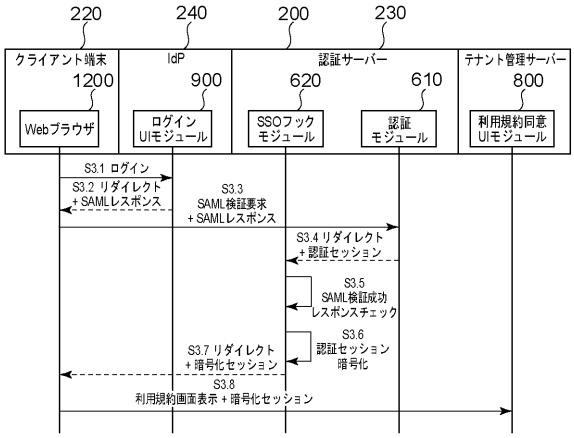
【図７】


S1.10の処理
S2.1 ユーザーは管理者?
S2.2 ユーザーにライセンスのロールが割り当てられている?
S2.3 ユーザーに割り当てられたロール
S2.4 対応する利用規約が同意済み?
S2.5 ログイン不可
S2.6 同意が必要な利用規約なし
S2.7 同意が必要な利用規約あり
S2.10 テナントに販売されたライセンス
S2.11 対応する利用規約が同意済み?
S2.12 同意が必要な利用規約なし
S2.13 同意が必要な利用規約あり
Yes
No
End

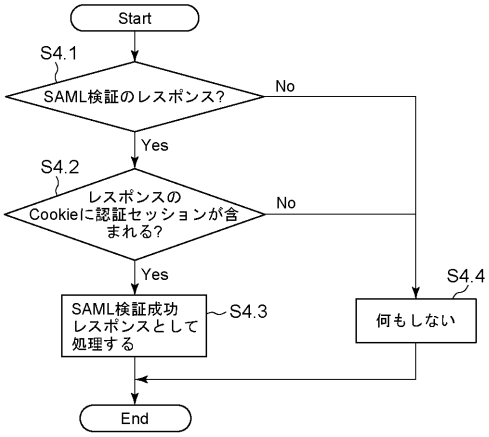
【図８】


8000
クラウドサービス
印刷サービス利用規約
次の利用規約を注意深くお読みください。
印刷サービス利用規約
印刷サービス利用規約zzです。xxに同意…
第1条（本契約の成立、効力及び終了）
…
…
8001
同意します
8002
(a)
8010
クラウドサービス
印刷サービス利用規約
次の利用規約を注意深くお読みください。
印刷サービス利用規約
印刷サービス利用規約zzです。xxに同意…
第1条（本契約の成立、効力及び終了）
…
…
8001
8012 同意します 同意しません 8013
(b)

【図９】



【図１０】



【図１１】

1300

テンポラリセッション管理テーブル

1301 テンポラリセッション	1302 認証セッション
TempSessionABCD	XXXX
TempSessionDCBA	YYYY