NTTソフトウェア

J-SaaSにおけるSAML2.0の適用

平成21年11月6日

NTTソフトウェア株式会社
モバイル＆セキュリティ・ソリューション事業グループ
永野　一郎

目次

1. J-SaaSについて

1．1 J-SaaSとは

J-SaaSとは？

経済産業省が推進するプロジェクト

「**中小企業向けSaaS活用基盤整備事業**」により構築されたSaaS活用型サービス

目的

中小企業のIT化支援および**電子申請の活用**を図るため、財務会計・経理・給与計算等の様々なサービスをワンストップで利用できる仕組みを提供すること

特徴

① SaaSの活用により、簡単に安心して利用できる環境を素早く提供できる

② 税理士、ITコーディネータ等の専門家によるサポートを提供している

③ 利用状況に応じ、民間事業者によるサービス継続の形態へ移行する予定

詳細は http://www.j-saas.co.jp 及び http://www.j-saaskensyu.jp/ を参照

1．2　経緯

主要なできごと

H21

◀3/31　サービス開始
18事業者、24サービス

◀6/25　経済産業省の定める
「SaaS向けSLAガイドライン」に準拠

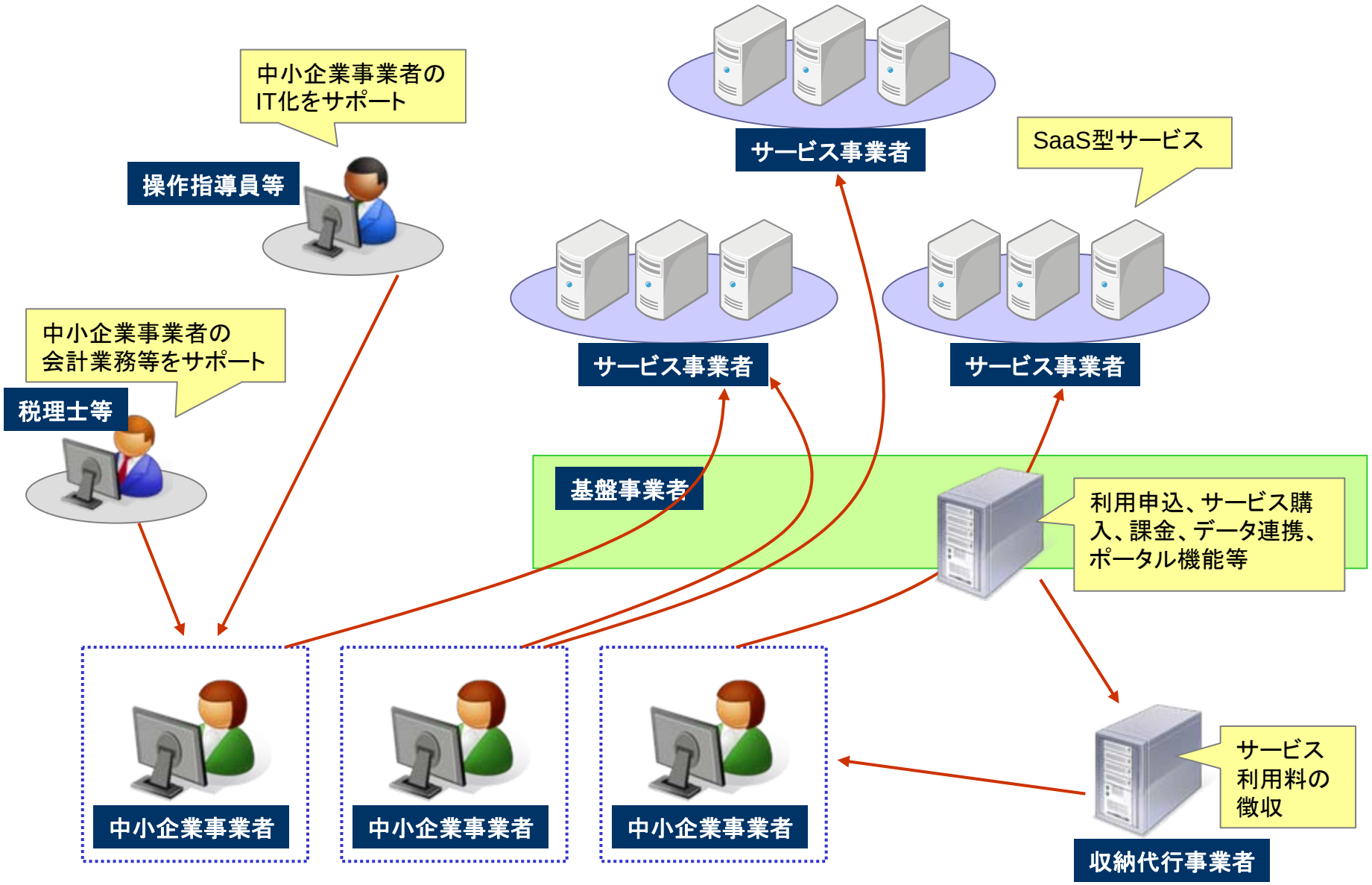
◀11月現在、
24事業者、32サービスまで拡大

H22

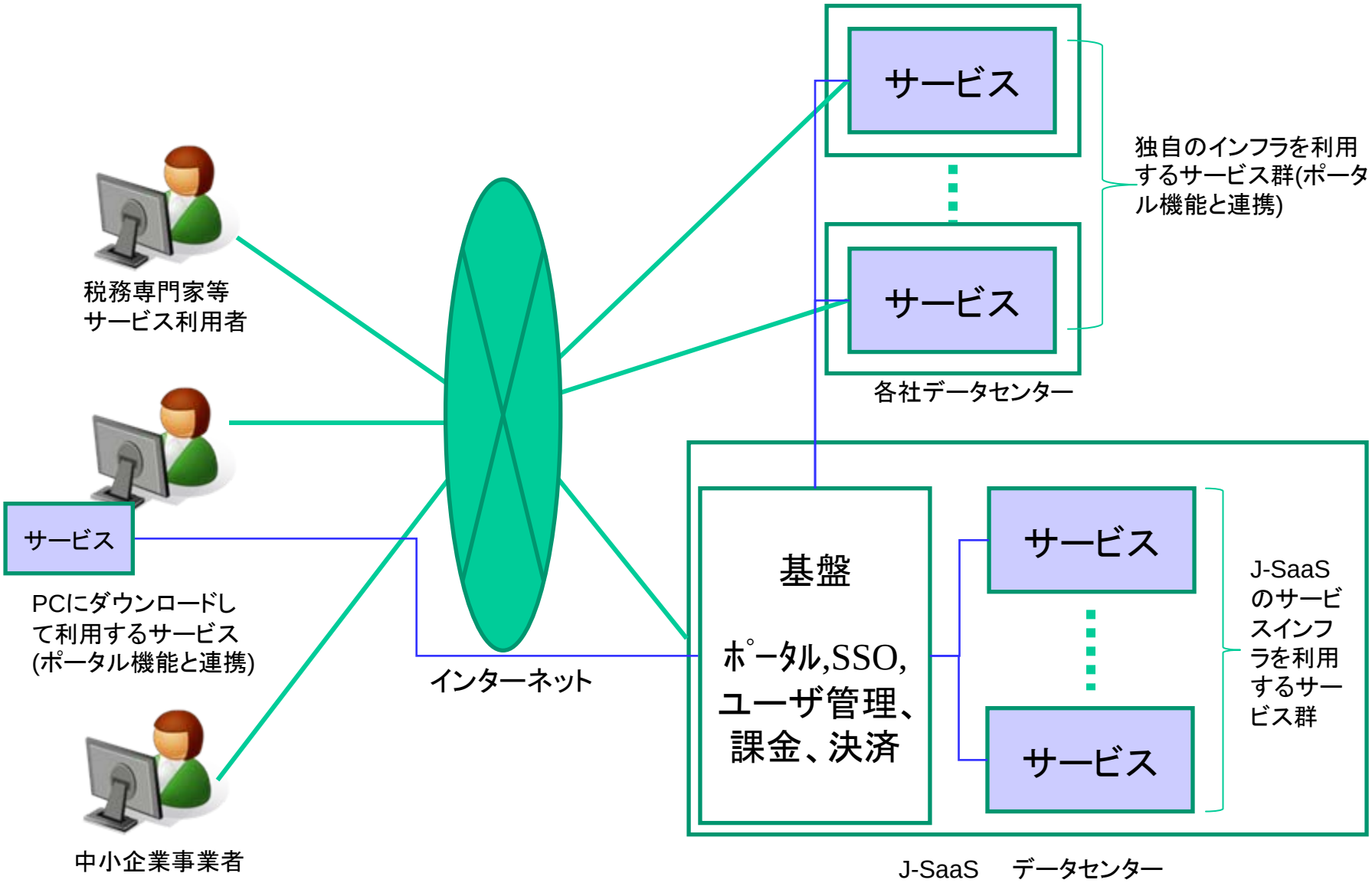
（予定）
民間企業による基盤運営の継続

※その他、J-SaaSの普及や情報提供、実機操作を目的として中小企業等に対して行われているJ-SaaS研修には、H21/10時点で13000名を超える利用者が受講されています。

1．3　登場人物



1．4　システムイメージ



1．5　基盤機能

J-SaaSの基盤は、事業者等のサービスに対し以下の機能を提供しています。

基盤の役割	提供機能
EC	サービスの検索、ライセンス販売
ポータル	周知、情報提供、利用申し込み 利用者の管理、利用サービスの管理
課金・請求	利用料金の管理、決済手段に応じたとりまとめ
SSO	ユーザ認証、各サービスへのシングルサインオン(認証連携)
データ連携	サービス間のデータ交換、基盤からサービスへの利用者・ライセンス情報の提供
収納代行	利用者への請求・入金管理

2. J-SaaSにおける認証機能

2．1　J-SaaSの認証機能の特徴

機能	概要
シングルサインオン	➢ サービスはシングルサインオンの連携を必須とする。 ➢ 利用者が基盤のIDを用いて基盤にログインした後は、任意のサービスを個別のログインなしで利用できる。
ID情報管理	➢ 企業/利用者IDの登録、ライセンス管理などSaaS独特の機能は基盤で提供する。 ➢ IDの連携はシステムによる連携と手作業による連携の2種類を用意。 ➢ 既に存在するIDはそのまま利用可能とする。
アグリゲーション	中小企業事業者サポートのため、税理士等がエンドユーザに成り代わって操作を代行することができる。
認証手段	➢ エンドユーザは利便性を考慮してIDとパスワードによる認証を行う。 ➢ 一方、税理士等は広範な利用権限を持つことからICカードを用いることとする。

2．2　シングルサインオン機能

3種類のSSO方式

様々なサービスがSSOを実現できるよう、基盤において3種類の方式をサポートした。

(1) エージェントモジュール方式

基盤と同じドメインに所属するWeb型サービスを対象に、CookieによるSSOを実現する。基盤より提供されるエージェントモジュールを利用する。

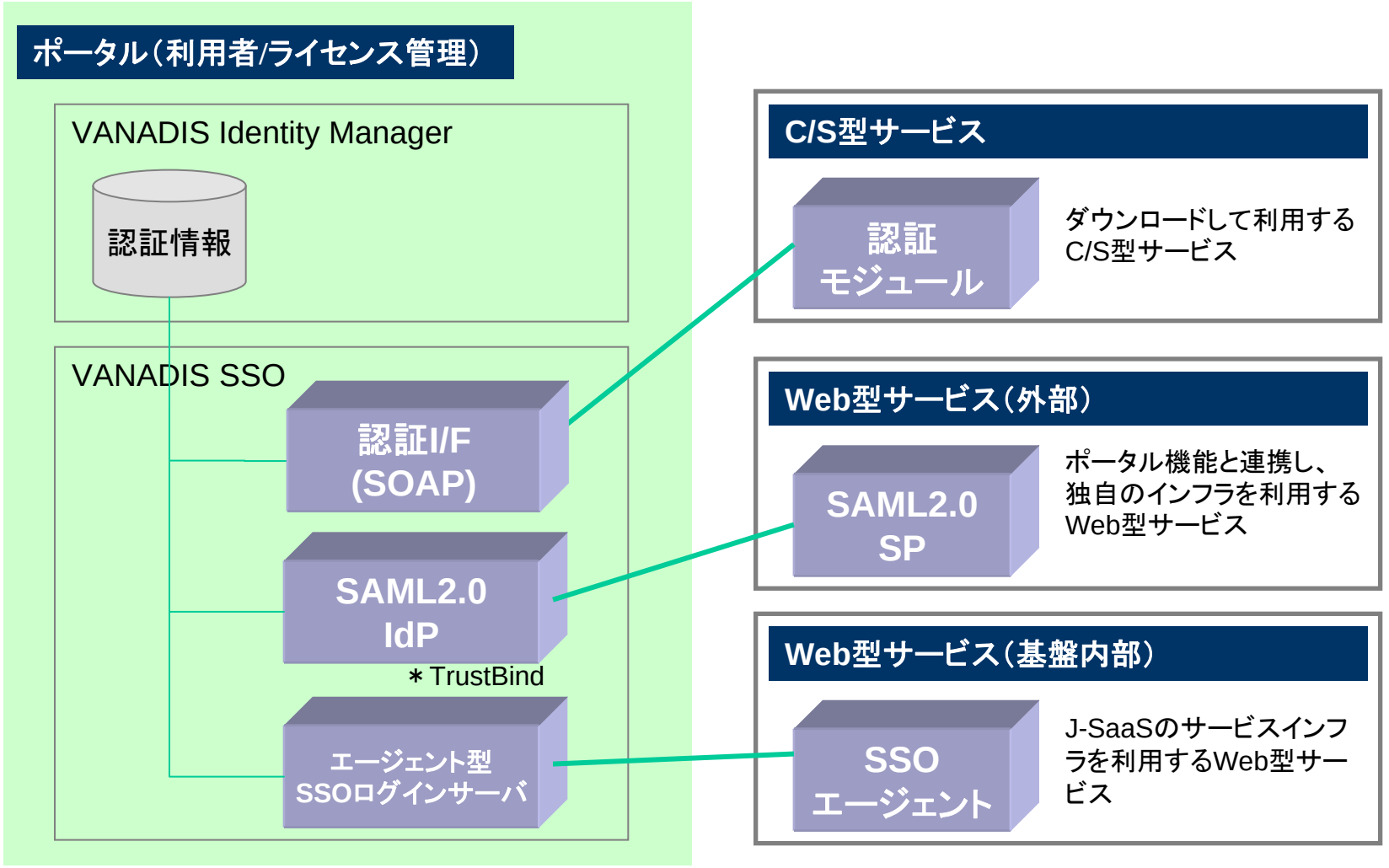
(2) SAML2.0方式

基盤と異なるドメインに所属するWeb型サービスを対象に、SAML2.0によるSSOを実現する。サービスへはI/F仕様を開示し、各サービスにて実装する。

(3) クライアント埋め込み認証モジュール方式

クライアント/サーバ型のサービスを対象に、クライアント側に埋め込む認証用モジュールを提供する。

2．3　基盤の認証機能構成



2．4　ID体系の考え方

■サービス事業者のAP改修負担の軽減とセキュリティ担保の観点から、基盤と各サービスのID体系は相互に独立したものとした

■両IDの紐付けは基盤が実施し、SSOのI/Fを介してサービスへ提供する
※なお、両IDともに企業をあらわす企業IDと利用者IDの組み合わせの形式

■アグリゲーション用に2種類のIDを利用

	Common-ID（共通ID）	Acting-ID（操作者ID）
用途	J-SaaSへのログイン用ID 基盤上での管理単位	各サービス事業者がサービス提供に利用 アグリゲーション時には被代理者のIDを指定
ID体系	基盤で定義	各サービスで定義
有効範囲	基盤全体でユニーク	サービス単位でユニーク
例	U1234（一般ユーザ） T2234（税理士）　等	123456 nagano　等

3. J-SaaSへのSAML2.0の適用

3．1　適用時の課題:Federation

課題

SAMLによるSSOに必要なFederationを、どのように実施するべきか

→ ユーザ自身により実施させるか？　システムで解決するか？

■ユーザ自身にFederationを実施させる場合、

①連携対象の各サイトで認証させる必要があり、ユーザの心理的負荷が高い

②連携対象サイトが増えるごとにFederation操作が必要となってしまう

■システムで解決するとした場合、

既にユーザが存在する場合、連携対象の特定が困難

■解決方法

SSO導入による操作の違いをユーザに意識させたくないという意図の下、Federationはシステム間で自動的に実施することとした。

- ユーザへのライセンス払い出しを契機に、基盤よりFederationを実施する
- 仮名には利用者IDそのものを採用

3．2　適用時の課題：ユーザ強制切替

課題

一旦ログイン/SSOした後に、IDを別のIDに切り替えることが可能か

■サービス仕様上、ポータルにログインしてSSOを実行した後に、異なるIDに切り替える機能が求められることになった。

→　ログアウトして再ログインする、あるいはブラウザを閉じれば実現可能だが、サービスの継続性が失われてしまう。

■解決方法

SAML2.0の以下の機能を利用

★AuthnRequest の ForceAuthn パラメータ

trueの場合、IDPは、既に認証済みのユーザであっても再度認証を実施させなければならない

→ 再認証時に切り替え対象のID/パスワードを入力させる

3．3 SAML2.0 I/F仕様

SAML2.0によるSSOで利用する仕様は、セキュリティ、接続作業の難易度などを考慮したうえで基盤側で定義し、サービス側へ接続仕様として提供。

利用可能アサーション	○：認証アサーション（Authentication Assertion） ×：属性アサーション（Attribute Assertion） ×：認可アサーション（Authorization Decision Assertion）
認証要求を行う場合のユーザの指定方法 (saml2md:NameIDFormat)	Persistent（urn:oasis:names:tc:SAML:2.0:nameid-format:persistent）
認証アサーションの有効期限	60,000ミリ秒前～認証アサーション設定時～300,000ミリ秒後
ForceAuthn	“true”必須
アサーションの署名 認証要求の署名	利用可能（強く推奨）
電文のバインディング	認証要求：HTTP-Redirect / HTTP-POST 認証アサーション：HTTP-POST

3．4　SAML2.0利用時のSSO

■アカウント作成・ライセンス割り当て時の流れ

1. 企業管理者が利用者のIDを登録
2. 企業管理者がサービスのライセンスを購入し、利用者へ割り当て
3. ライセンスの割り当て情報を元に、基盤側でIdP、サービス側でSPとしてのフェデレーションを事前に実施。仮名は利用者のIDの一つを利用。

■サービス利用時

1. ユーザはVANADIS SSOのｴｰｼﾞｪﾝﾄ型SSOを利用して基盤ポータルへログイン（Cookieとしてアクセスチケットが発行される）
2. ユーザがサービスへアクセスすると、SPはIdPに認証要求を送付する。
3. IdPは、基盤へログイン時に発行したCookie内のアクセスチケットとポータル内の情報を元に、認証アサーションを作成。
4. SPは受け取った認証アサーションを元に、サービス内での認証を実施。

3．5　サービス事業者による接続作業

1次募集のサービス事業者については、全サービス無事に接続し、サービスを開始。

★サービス事業者とのSAML2.0による接続過程で生じた問題

①メタデータの設定誤り

AssertionConsumerService#Bindingに使用できない値が設定

②時刻同期が取れていなかったためSSOに失敗

Assertion/Conditions#NotBeforeから

Assertion/Conditions#NotOnOrAfterの期間を超えてずれていた

→ 不正なAssertionとしてエラー

③電文解析にはスキルが必要

(i) 正しいSAML2.0メッセージが交換されていたにも関わらず、

誤っていると勘違いしてのお問合せがあった

(ii) SSOは行われているが、毎回Federation実施のログが出ていた

→ AuthnRequestに毎回AllowCreate=trueが設定されていた

4. まとめ

4．1　総括

■J-SaaSとはSaaSを活用した中小企業向けワンストップサービス

→ ***日本における先駆的なSaaS基盤***

■サービス向けに各種機能を基盤として提供。その一つとしてSSO機能を用意

■SSOにはアプリケーションの種類、配置に応じて3種類の方法を用意
主に外部にあるWeb型サービスのためにSAML2.0を採用。

■弊社が果たした役割

- 基盤へのプロダクト提供（IdP)
- 一部サービス事業者へのプロダクト提供（SP)
- 一部サービス事業者へサービスと基盤の接続支援

4．2 SAML2.0のJ-SaaSへの適用

■**J-SaaSにおいてはSAML2.0が持つ以下の特徴が有効であった**

- Federationによる共通IDとAP側IDの連携が可能
- 認証を柔軟に制御できる仕組み（ForceAuthn、AuthnContextなど）が存在
- AttributeConsumingServiceIndexによる属性交換の仕組み
- プロダクトに依存しない接続性
- SaaSに求められる高いセキュリティ要件への適合

■**適用作業について**

- サービスをSAMLに適用する作業はそれほど困難ではない
- ただし、接続においてトラブルが発生した場合、トラブルシュートにはSAMLに関するノウハウが必要

4．3 Kantara Initiativeへの期待

■認証関連

- SAML2.0とOpenIDの相互接続性を一層高めてほしい
- プロトコルが混在する環境について、信頼性やセキュリティレベルを評価する基準、仕組みを確立してほしい

■ID管理一般

- 複数システム間でIDプロビジョニングの同期を図る仕組みはないか？
- SaaS/クラウドでの利用シーンをモデル化できないか？

ご清聴ありがとうございました。