XML Consortium

最新XMLセキュリティ技術概要

2005年12月15日

XMLコンソーシアム セキュリティ部会
横溝 良和（キヤノン株式会社）
山根　利夫（株式会社日立製作所）
中山 弘二郎（株式会社日立製作所）
西村利浩（富士通株式会社）

アジェンダ

- XML Security関連規格の調査
- SOX法対策とアクセス制御
  XACML v2.0の概要
- Webサービスポリシー概要
- WS-Trust概要

XML Security関連規格の調査

2005年12月15日

XMLコンソーシアム セキュリティ部会
横溝良和（キヤノン株式会社）

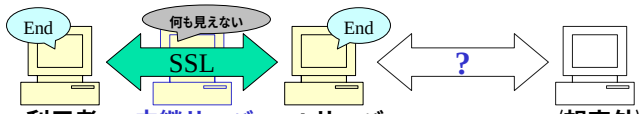
WS-Security（WSS）の目的

- Webサービスをセキュアにするために、その基本プロトコルであるSOAPを機能拡張する標準（WSS）を作り、End-to-Endの完全性と秘匿性を実現する。
  - Webサービスのメッセージ通信が、複数のポイントを経由して行われる場合でも、セキュアな通信サービスが確保される。
- 既存のセキュリティ技術を統一的に使う仕組みを提供する。
- 幅広いセキュリティ・モデルのサポート
  - 複数のセキュリティ・トークン形式
  - 複数の信頼ドメイン
  - 複数の署名形式
  - 複数の暗号技術

SSLとWSSの違い

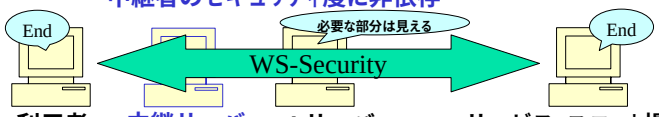
SSL：Point-to-Point

トランスポート層のセキュリティ
WS中継者から先のセキュリティが不明



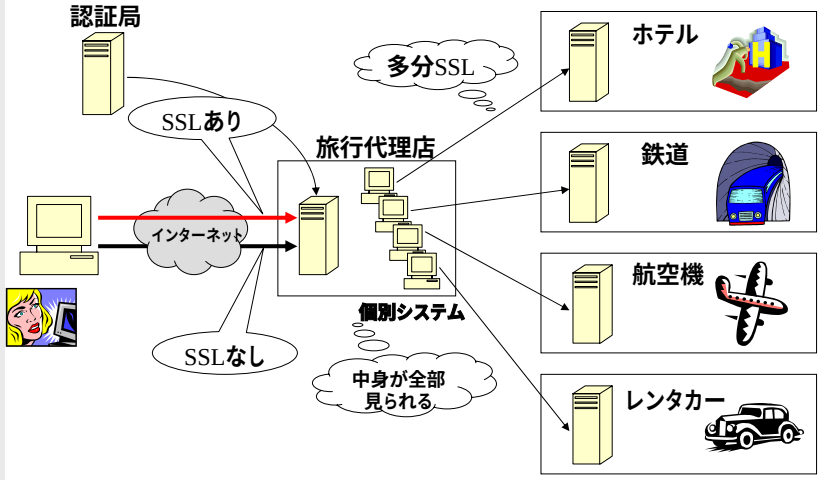
WSS：End-to-End

メッセージ・コンテンツのセキュリティ
中継者のセキュリティ度に非依存



参考：岡村和英著「Webサービスのセキュリティ」

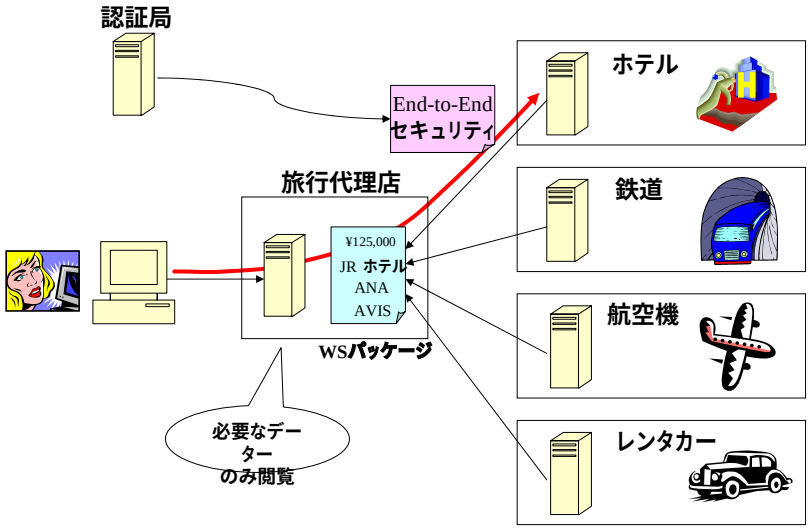
SSLの利用例



旅行代理店から先の通信のセキュリティが保証されない。同代理店に、全ての情報が見られてしまう。

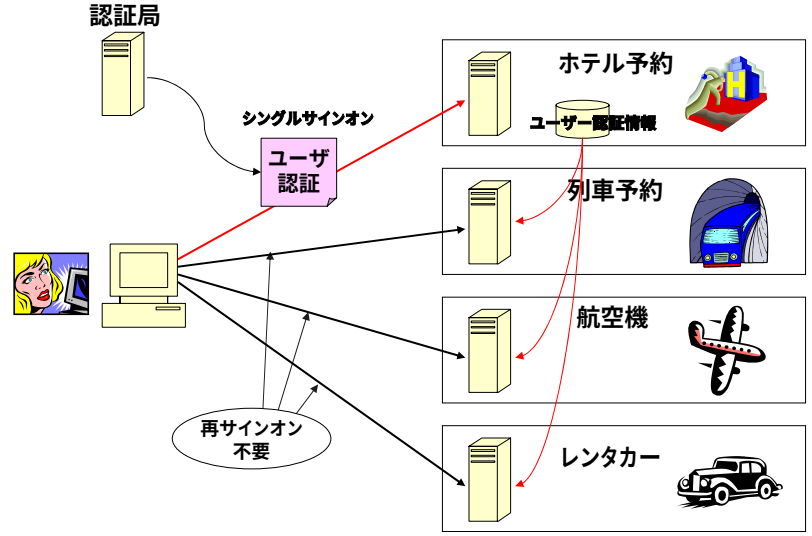
2005.4.27Y.Yokomizo

WSSの利用例



旅行代理店から先の通信のセキュリティが保証され、同代理店には、必要最小限の情報しか開示しなくて良い。

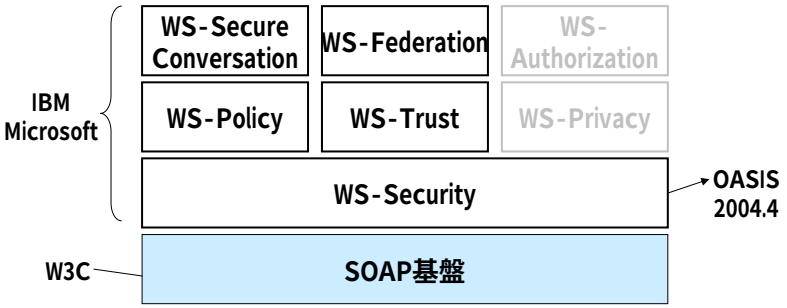
SAMLの利用例



一度認証されれば、他の関連サービスは認証無しで利用できる。

IBM＋Microsoftの共同提案（2002.4.7）

「Webサービスのセキュリティ: アーキテクチャとロードマップの提案」

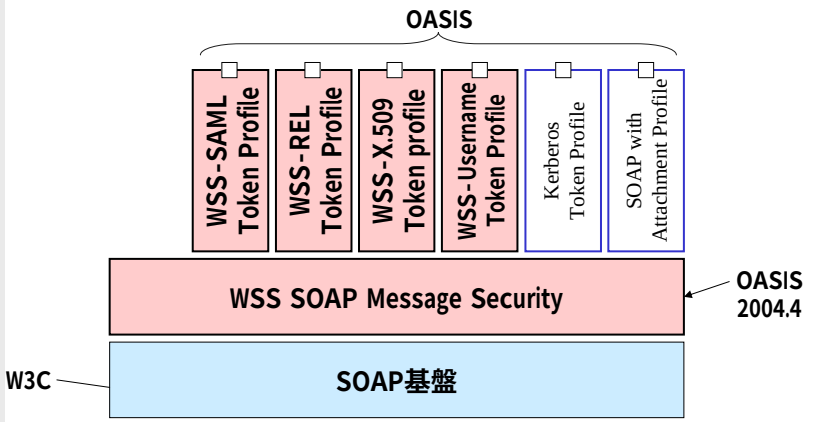


WS-Security	SOAPメッセージへの署名ヘッダ、暗号化ヘッダ、セキュリティトークン添付方法
WS-Policy	中間ノードやエンドポイントでのセキュリティポリシーの権限や制約の記述
WS-Trust	Webサービスを安全に相互運用する為の信頼モデルのフレームワーク
WS-Privacy	プライバシーの設定、実践ステートメントを提示する方法のモデル
WS-SecureConversation	セキュリティ コンテキストの交換、セッション鍵確立と派生などの、メッセージ交換を管理、認証する方法
WS-Federation	アイデンティティの連合など、異種の連合された環境における信頼関係の管理と仲介の方法について
WS-Authorization	認可データや認可ポリシーを管理する方法

WS-SecurityはOASISの標準になった

引用文献: IBM Corporation と Microsoft Corporation 共著によるホワイト ペーパー

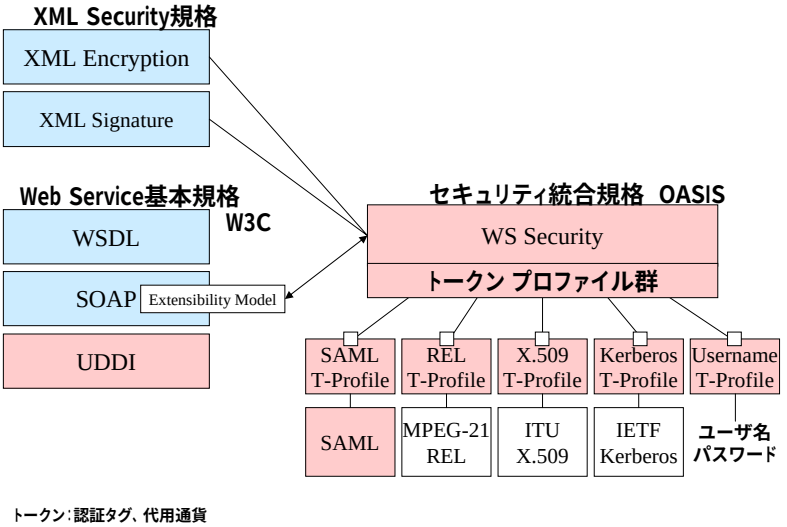
OASIS WS-Security規格群（2005.5）



WS-SecurityはOASISの標準になった

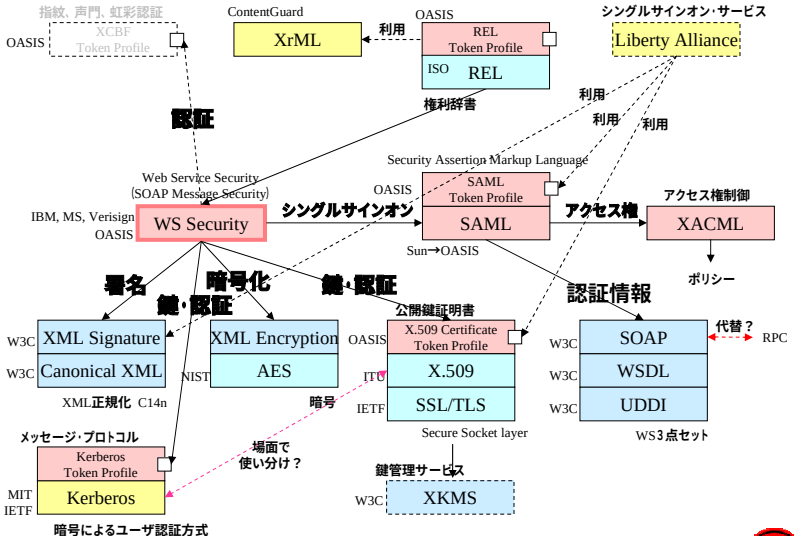
2005.4.27Y.Yokomizo

規格としての位置付け



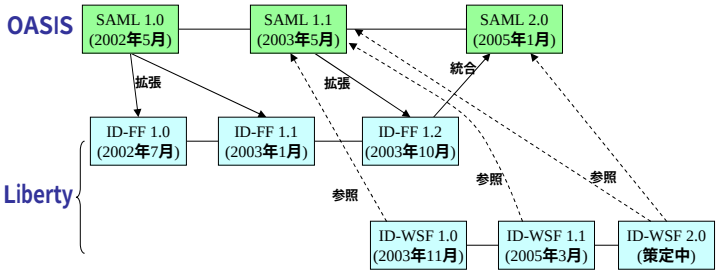
トークン：認証タグ、代用通貨

セキュリティ規格のマップ



Liberty AllienceとOASISの関係

SAMLとLiberty ID-FF, ID-WSFの関係



引用：XMLコンソーシアム主催LibertyAlliance講演資料を追加修正

Liberty Allianceは、「シングル・サインオン」を主要な応用事例とする、ID（アイデンティティ）管理の仕組みを標準化している標準化団体である。 Libertyは、OASISのSAML 1.0をベースに、ID-FF (Identity Federation Framework)を開発したが、ID-FF 1.2の段階で仕様をSAML 2.0に組み入れ、標準化作業はOASISに移管した。一方、ID-WSF (Web Service Framework)とID-SIS (Service Interface Specifications) については引き続きLiberty Allianceが開発を進めるという。ID管理アーキテクチャーは、オープンな連携型の分散ID管理手法を提唱している。アイデンティティ連携（ID-FF）とは、アカウントのサービス間連携とシングルサインオン（SSO）の事である。テストにより互換性が確認された製品には、「互換性認証ロゴ」の表示を認めるという。動作環境はWebサービスのSSOを対象とする。この分野での「ディスカバリーサービス」とは、個人情報のディスカバリーの事である。

XML Consortium

SOX法対策とアクセス制御

XACML v2.0の概要

eXtensible Access Control Markup Language

OASIS Standard, 1 Feb 2005

2005年12月15日
XMLコンソーシアム セキュリティ部会
山根　利夫 (株) 日立製作所

アジェンダ

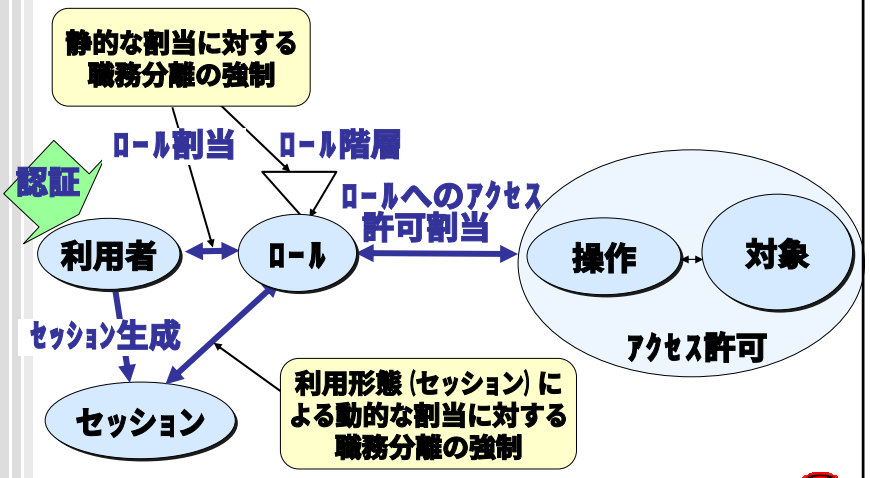
- 全体統制としてのｱｸｾｽ制御
- ﾛｰﾙﾍﾞｰｽｱｸｾｽ制御
- XACMLの概要
- V2.0の強化点と製品化状況

1. 全体統制としてのｱｸｾｽ制御

- 「職務分離」と「ｱｸｾｽ制御」は、SOX法対応に於いて、ｼｽﾃﾑを保障する「IT全体統制」の基本要素。（日本版SOX法　2006年？）
  - ｼｽﾃﾑへ利用者を登録する方式での問題点
    - ｼｽﾃﾑ毎／職務毎の利用者登録
    - 充分に複雑なﾊﾟｽﾜｰﾄﾞ／定期的なﾊﾟｽﾜｰﾄﾞ変更
    - 退職、職務変更等に対する保守
  - IT全体統制としてのﾛｰﾙﾍﾞｰｽｱｸｾｽ制御
    - 利用者認証とｱｸｾｽ制御の分離
    - 職位等、利用者の属性／役割（ﾛｰﾙ）による自動的なｱｸｾｽ権の保守

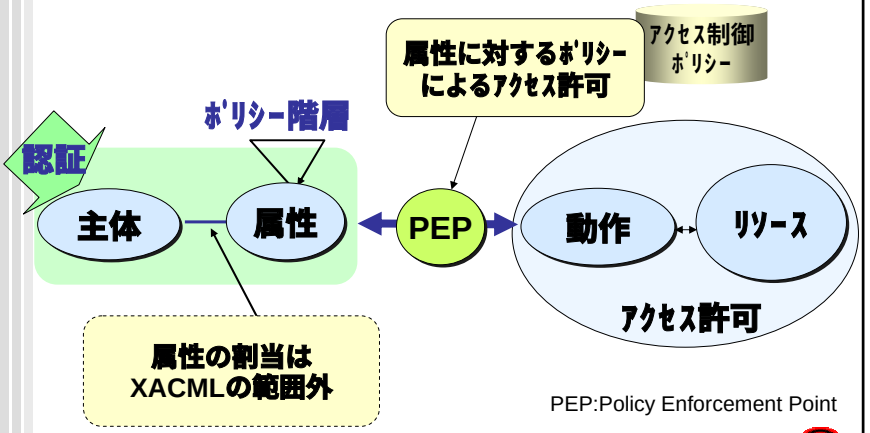
2. ロールベースアクセス制御：RBAC

Role-Based Access Control　ANSI INCITS 359-2004. 2004/3



3. XACMLの概要

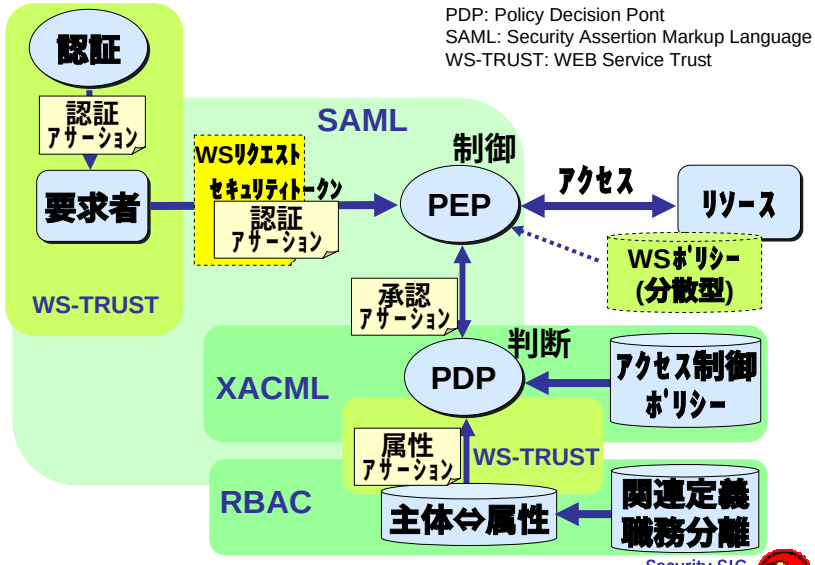
- 大組織では「利用者」と「ロール」の関連管理が膨大
- XACMLでは、既存の組織管理上の「属性」情報を活用



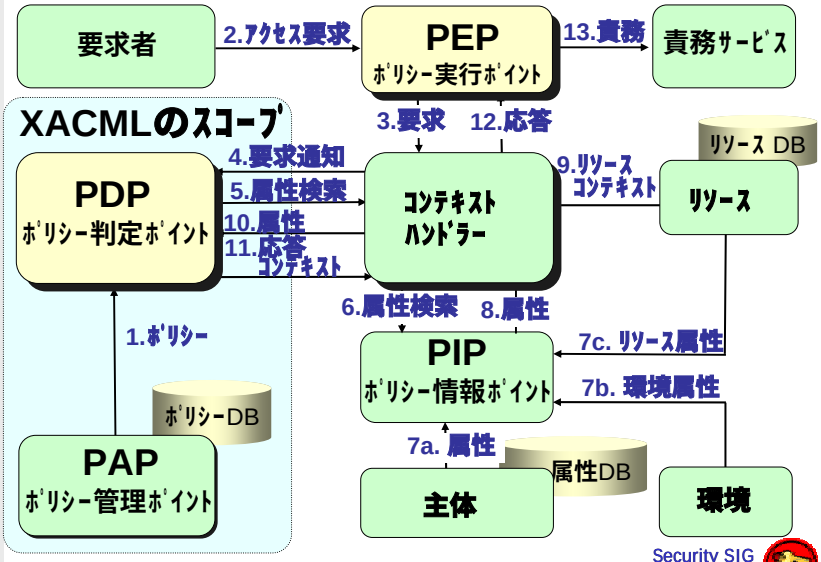
PEP:Policy Enforcement Point

3.1 XACMLの位置付け

PDP: Policy Decision Pont
SAML: Security Assertion Markup Language
WS-TRUST: WEB Service Trust



3.2 XACMLのデータフロー詳細



3.3 ポリシー定義例

```
<Policy PolicyId="Policy1"  RuleCombiningAlgID="Deny-Override">
   <Rule RuleID="Rule1" Effect=Permit>
      <Target>
         <Subjects>
            <Subject>
               <SubjectMatch MatchId="function:string-match">
                <SubjectAttributeDesignatir  AttributeId="identifier:subject:role"/>
                <AttributeValue>administrator</AttributeValue>
            </Subject>
         </Subjects>
         <Actions>
            <Action>
               <ActionMatch  ActionMatchId="function:string-equal">
                  <TargetAttributeDesignator AttributeId="urn:xx:target:action"/>
                  <AttributeValue>read</AttributeValue>
               </ActionMatch>
            </Action>
         </Actions>
      </Target>
   </Rule>
    :
</Policy>
```

アクセスを許可する

何れかのruleで否なら不許可

roleがadministratorで

actionがreadの場合

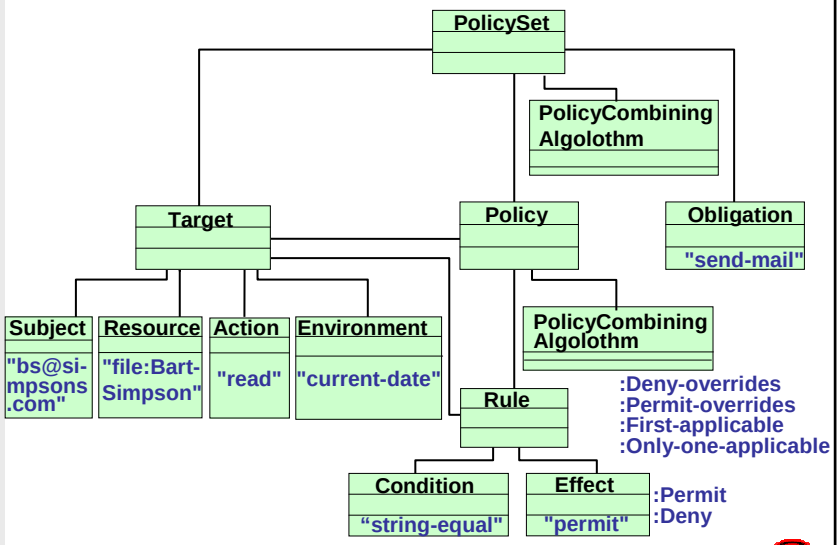
4. V2.0の強化点と製品化状況

- V2.0での主な強化点
  - プロファイルの整備（SAML,Digital Signature, LDAP 等）
  - ﾌﾟﾗｲﾊﾞｼｰポリシー対応
  - 複数ﾘｿｰｽ、複数の属性値対応
- 製品化状況
  - SUN Microsystems

    ’05/1　SUNXACMLでXACMLv2.0をサポート済み。
  - ORACLE

    買収したOblix製品をﾍﾞｰｽに、Identity Managementで’06半ばに、XACMLをサポート予定。

付録1　ﾎﾟﾘｼｰ言語仕様



XML Consortium

Webサービスポリシー概要

(WS-Policy, WS-PolicyAttachment, WS-SecurityPolicy)

2005年12月15日
XMLコンソーシアム セキュリティ部会
中山 弘二郎（株式会社 日立製作所）

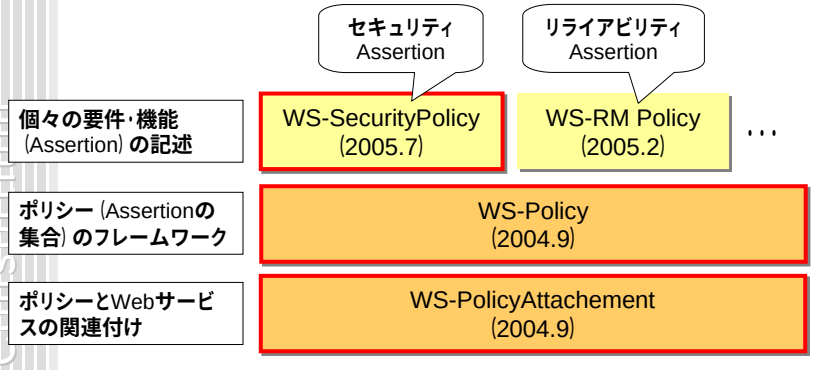
Webサービスポリシーの概要

- Webサービスのポリシーとは
  - Webサービスの要件や機能
    - 例1）WebサービスAを利用するためには、SOAPリクエストに署名がされていなければならない
    - 例2）WebサービスBはAESアルゴリズムによる暗号化をサポートしている
  - WSDLでは記述されないが、Webサービスを利用するためには必要な情報
- ポリシーの利用例
  - Webサービスの要件や機能をポリシーとして記述し公開
  - クライアントは、送信メッセージがポリシーを満たすように処理
  - Webサービスは、受信メッセージがポリシーを満たしているか検証

ポリシーの標準仕様を策定することで、Webサービスの相互接続性向上が期待できる

SOAP：Simple Object Access Protocol　AES：Advanced Encryption Standard
WSDL：Web Services Description Language

ポリシー関連仕様



：OASISに提出済みの仕様
：標準化団体未提出の仕様

（）内は最新版の仕様書の公開月

WS-RM：WS-ReliableMessaging

WS-SecurityPolicy概要

- WS-SecurityPolicyとは
  - Webサービスのセキュリティに関するAssertionの記述方法を規定した仕様
  - WS-Security、WS-Trust、WS-SecureConversation、トランスポートレベルセキュリティに関するAssertionの記述方法を規定

◆ **Assertionの例**

```
<sp:EncryptedParts>
  <sp:Body/>
</sp:EncryptedParts>
```

SOAPボディの秘匿性が確保されていなければならない

メッセージには常にUsernameTokenが含まれていなければならない

```
<sp:UsernameToken
  sp:IncludeToken=".../IncludeToken/Always" />
```


WS-Policy概要

- WS-Policyは、ポリシー表記のフレームワークを規定した仕様

◆ **ポリシー表記の例**

```
<wsp:Policy>
  <wsp:ExactlyOne>
    <wsp:All>
      Assertion A
      Assertion B
    </wsp:All>
    <wsp:All>
      Assertion C
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
```

※ Normal Formで記述した場合

Policy Expression

ポリシーの表記。Alternativeの集合。ポリシーを満たすには、下位のAlternativeのどれか一つを満たす必要がある

Policy Alternative

Assertionの集合。Alternativeを満たすには、下位のAssertionをすべて満たす必要がある

Policy Assertion

個々の要件や機能の表記。Assertionのスキーマは別仕様で規定する

WS-PolicyAttachment概要

- WS-PolicyAttachmentは、ポリシーとWebサービスを関連付ける方法を規定した仕様

◆ **WSDLへの関連付けの例**

WSDL

```
<wsdl:definitions>
  <wsp:UsingPolicy>
  <wsdl:message
    wsp:PolicyURIs="http://example.com/wsp#ABC" />
  ...
</wsdl:definition>
```

WS-PolicyAttachmentを使っていることを宣言

URIによりポリシーを参照

ポリシー

```
<wsp:Policy
    xml:base="http://example.com/wsp"
    wsu:Id="ABC">
  ...
</wsp:policy>
```

ポリシーのURIを定義

まとめ

- Webサービスのポリシーとは、Webサービスの機能や要件のこと
- 以下のポリシー関連仕様が提案されている
  - 個々の機能・要件の記述
    - WS-SecurityPolicy
    - WS-RM Policy
  - ポリシーのフレームワーク
    - WS-Policy
  - ポリシーとWebサービスの関連付け
    - WS-PolicyAttachment
- 今後、ポリシー仕様が標準化されることで、Webサービスの相互接続性の向上が期待できる

参考文献

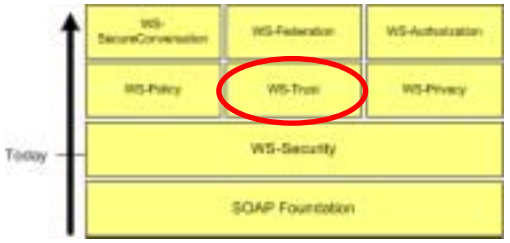
- **WS-Policy**
  (Web Services Policy Framework)
  http://msdn.microsoft.com/ws/2004/09/policy/
- **WS-PolicyAttachment**
  (Web Services Policy Attachment)
  http://msdn.microsoft.com/ws/2004/09/policyattachment/
- **WS-SecurityPolicy**
  (Web Services Security Policy Language)
  http://msdn.microsoft.com/ws/2005/07/ws-security-policy/

WS-Trust概要

2005年12月15日
XMLコンソーシアム セキュリティ部会
西村利浩（富士通株式会社）

WS-Trust: 経緯

- これまでの経緯
  - 初出：IBMとMicrosoftによるホワイトペーパー「Security in a Web Services World: A Proposed Architecture and Roadmap」(2002年4月)



(「Security in a Web Services World: A Proposed Architecture and Roadmap」より)

  - 2002年12月にVersion 1.0、2004年5月にVersion 1.1、2005年2月にVersion 1.2
  - 2005年12月～OASIS WS-SX TCで標準化

WS-Trust: 概要(1)

- Webサービスの信頼モデル
  - Webサービスは、メッセージを受け取る際に、クレーム(名前、鍵、許可など)の証明を要求できる

WS-SecurityPolicyで記述

  - クレームはセキュリティ・トークンで表される

リクエスタ → Webサービス

ポリシー(例):
SAML属性アサーションによる年齢の証明が必要

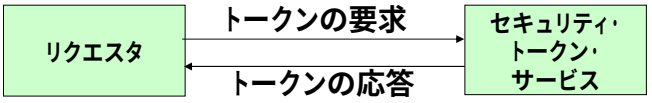
```
<sp:IssuedToken ...>
    <sp:Issuer>...</sp:Issuer>
    <sp:RequestSecurityTokenTemplate>
        <wst:TokenType>...SAML:1.1:assertion</wst:TokenType>
        <wst:Claims ...>...</wst:Claims>
    </sp:RequestSecurityTokenTemplate>
</sp:IssuedToken>
```

WS-Trust: 概要(2)

- セキュリティ・トークンはセキュリティ・トークン・サービス(STS)から取得

WS-Trustを利用

```
<wst:RequestSecurityToken Context="...">
    <wst:TokenType>...SAML:1.1:assertion</wst:TokenType>
    <wst:RequestType>...Issue</wst:RequestType>
    ...
</wst:RequestSecurityToken>
```




```
<wst:RequestSecurityTokenResponse Context="...">
    <wst:TokenType>...SAML:1.1:assertion</wst:TokenType>
    <wst:RequestedSecurityToken>
        <saml:assertion>...</saml:assertion>
    </wst:RequestedSecurityToken>
    ...
</wst:RequestSecurityTokenResponse>
```


WS-Trust: 概要(3)

- トークンを提示しWebサービスを利用

WS-Security(WSS)を利用


```
<S:Envelope>
    <S:Header>
        <wsse:Security>
            <saml:assertion>...</saml:assertion>
            ...
        </wsse:Security>
    </S:Header>
    <S:Body>
        ...
    </S:Body>
</S:Envelope>
```

WS-Trust: 基本プロトコル

- 要求メッセージ RequestSecurityToken

```
<wst:RequestSecurityToken Context="...">
    <wst:TokenType>...</wst:TokenType>
    <wst:RequestType>...</wst:RequestType>
    ...
</wst:RequestSecurityToken>
```

トークンの型を指定

要求の型を指定

- 応答メッセージ RequestSecurityTokenResponse

```
<wst:RequestSecurityTokenResponse Context="...">
    <wst:TokenType>...</wst:TokenType>
    <wst:RequestedSecurityToken>...
                        </wst:RequestedSecurityToken>
    ...
</wst:RequestSecurityTokenResponse>
```

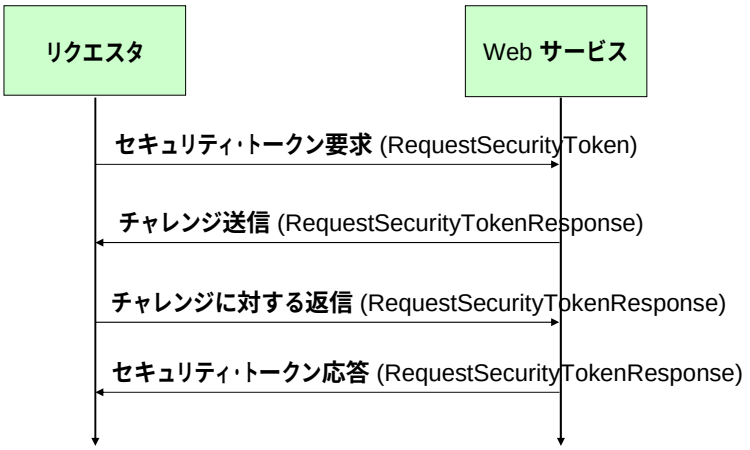
返却されるトークン

WS-Trust: 基本的な要求のタイプ

- 要求のタイプ
  - Issue – 要求において提供/証明されたクレデンシャルに基づいて、(場合によっては新しい証明情報とともに) 新しいトークンが発行される。
  - Renew – 以前発行された有効期限付きのトークンを提示することにより、 新しい期限で同じトークンが返される。
  - Cancel – 以前発行されたトークンが必要なくなったとき、トークンをキャンセルし、その利用を終了する。
  - Validate – 指定されたセキュリティトークンの有効性が評価され、その結果が返される。

WS-Trust: 交渉/チャレンジ

- 複数メッセージ交換による交渉/チャレンジも許す



WS-Trust: まとめ

- WS-Trustはセキュリティ・トークン・サービスとやりとりするためのプロトコルを規定
- 基本的な要求のタイプとして、トークンの発行、更新、取り消し、検証を規定
- 複雑な交渉/チャレンジのための拡張性も保持
- OASIS WS-SX TCで仕様の標準化が開始
  - WS-Trust
  - WS-SecureConversation
  - WS-SecurityPolicy

XML Consortium

最新ＸＭＬセキュリティ技術概要

2005年12月15日

XMLコンソーシアム セキュリティ部会
横溝 良和（キヤノン株式会社）
山根　利夫（株式会社日立製作所）
中山 弘二郎（株式会社日立製作所）
西村利浩（富士通株式会社）