XML Consortium

～第6回XMLコンソーシアムWeek～

Webサービスのセキュリティ規格の標準化動向

2007年5月21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）

アジェンダ

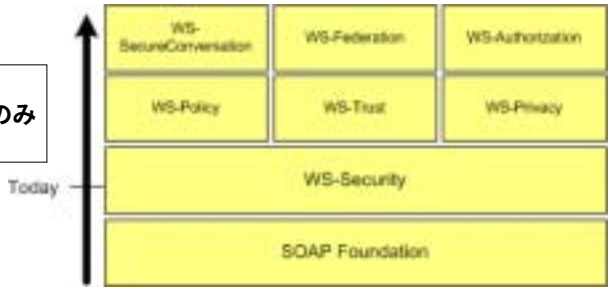
- 標準化動向
  - 2002年から現在に至る標準化の進展
  - 最近起こった問題点
- 仕様解説
  - WS-Security
  - WS-Policy
  - WS-PolicyAttachment
  - WS-SecurityPolicy

標準化動向

Webサービスのセキュリティ仕様(2002/4)

- 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)

この時点では
WS-Securityのみ
公開



「Security in a Web Services World: A Proposed Architecture and Roadmap」より
(http://msdn2.microsoft.com/en-us/library/ms977312.aspx)

2002年

1月　7月　12月

▲4月 ホワイトペーパー公開

▲4月 WS-Security 1.0公開

▲6月 WS-Security OASIS提出アナウンス

▲8月 WS-Security Addendum公開

OASIS Web Services Security (WSS) TCでWS-Securityの標準化を開始

▲7月 WSS TC 参加者募集

▲9月 最初のF2F

OASIS WSS TC

▲12月 F2F

12月 以下の仕様公開
- WS-Trust 1.0
- WS-SecureConversation 1.0
- WS-SecurityPolicy 1.0
- WS-Policy 1.0
- WS-PolicyAttachment 1.0
- WS-PolicyAssertions 1.0

WS-Policy関連仕様が細分化
このうちセキュリティ関連仕様は
WS-SecurityPolicy

続々と標準化が開始すると期待されたが・・・

F2F: Face to Face (電話会議ではなく)顔を合わせてのミーティング

2003年

1月　7月　12月

▲6月 F2F/Interop

9-10月 WSS 1.0公開レビュー

OASIS WSS TC

▲5月 WS-Policy 1.1 (1.01?)公開

▲5月 WS-PolicyAttachment 1.1 (1.01?)公開

▲5月 WS-PolicyAssertions 1.1公開

▲7月 WS-Federation 1.0公開

2004年

1月 7月 12月

▲3月 WSS 1.0 OASISメンバー投票
★4月 WSS 1.0 OASIS Standard
7-8月 WSS:SAML/REL公開レビュー
▲10月
WSS:SAML/REL OASISメンバー投票
★12月
WSS:SAML/REL OASIS Standard

OASIS WSS TC

▲5月 WS-Trust 1.1公開
▲5月 WS-SecureConversation 1.1公開

▲9月 WS-Policy 1.1?公開
▲9月 WS-PolicyAttachment 1.1?公開

OASISメンバー投票: OASIS Standardとして承認するためのOASISメンバーによる投票
REL: Right Expression Language

2005年

1月 7月 12月

1-2月 WSS:SWA公開レビュー
7-9月 WSS 1.1公開レビュー

OASIS WSS TC

▲2月 WS-Trust 1.2公開
▲2月 WS-SecureConversation 1.2公開

▲7月 WS-SecurityPolicy 1.1公開

OASIS Web Services Secure Exchange (WS-SX) TCでWS-Trust、WS-SecureConversation、WS-SecurityPolicyの標準化を開始

▲10月 WS-SX TC参加者募集
▲12月
WS-SX TC最初のF2F

OASIS WS-SX TC

最初のF2F後、18ヶ月以内に
Committee Specification完成を目標

SWA: SOAP Messages with Attachments
Committee Specification: OASISメンバー投票にかけるTCでの最終仕様

2006年

1月 7月 12月

▲1月 WSS 1.1 OASISメンバー投票
★2月 WSS 1.1 OASIS Standard
10月 WSS 1.1エラッタ公開レビュー
★11月 WSS 1.1 承認エラッタ
OASIS WSS TC

▲4月 WS-SX TC F2F
9-11月 WS-Trust 1.3, WS-SecureConversation 1.3公開レビュー
12-2月 WS-SecuritPolicy 1.2 公開レビュー
OASIS WS-SX TC

▲3月 WS-Policy 1.2公開
▲3月 WS-PolicyAttachment 1.2公開
▲4月 WS-Policy W3C Member Submission
▲4月 WS-PolicyAttachment W3C Member Submission

W3C Web Services Policy WGでWS-Policy、WS-PolicyAttachmentの標準化を開始

▲6月WS-PolicyのWorking Group設立
▲7月 F2F
▲11月 F2F
▲7月 First Public Working Draft
▲11月 Last Call
W3C WS-Policy WG

3年半振りに改版
Authorization, Privacyに関する部分も

▲12月 WS-Federation 1.1公開

Security SIG
21-May-2007

2007年

1月 7月 12月

▲1月 WSS TC終了

▲2月 WS-Trust 1.3,WS-SecureConversation 1.3 OASISメンバー投票
★3月 WS-SecureConversation 1.3 OASIS Standard
★3月 WS-Trust 1.3 OASIS Standard
11-2月 WS-SecurityPolicy 1.2公開レビュー
3-4月 WS-SecuritPolicy 1.2公開レビュー
9月以降? 改版
△6月 WS-SecurityPolicy 1.2 OASISメンバー投票
☆7月 WS-SecurityPolicy 1.2 OASIS Standard
OASIS WS-SX TC

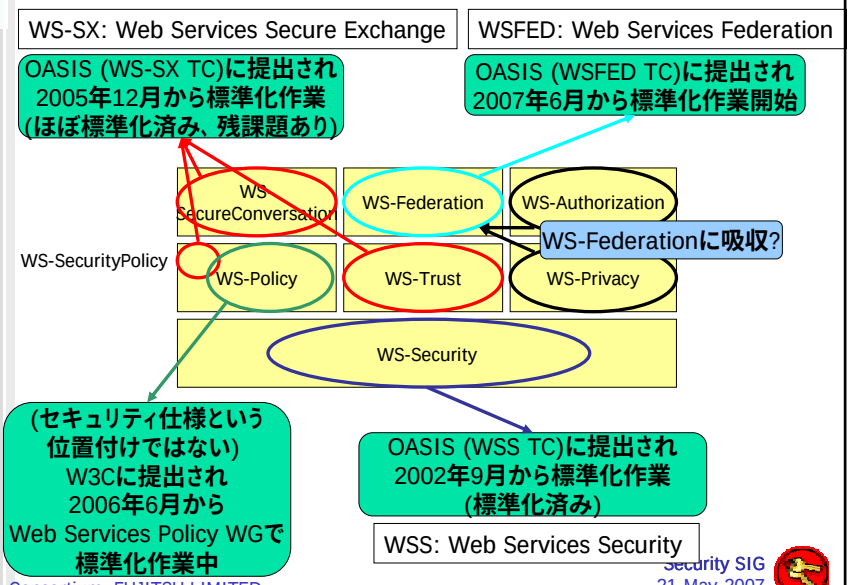
▲1月 F2F
▲3月 F2F
▲2月 WS-Policy 1.5 Candidate Recommendation (Call for Implementation)
▲3月 WS-Policy 1.5 Candidate Recommendation
△7月 Proposed Recommendation
☆8月 W3C Recommendation △12月 活動終了
W3C WS-Policy WG

OASIS Web Services Federation (WSFED) TCでWS-Federationの標準化を開始

▲3月 チャータ提案 △6月 最初のF2F
▲4月 参加者募集
最初のF2F後、18ヶ月以内にCommittee Specification完成を目標
OASIS WSFED TC

Security SIG
21-May-2007

Webサービスのセキュリティ仕様(2007/5)



WS-Trust OASISメンバー投票における問題

- WS-Trust 1.3のOASISメンバー投票(2007年2月後半)で**反対票**
- 理由:WS-Policy 1.2 W3C Member Submissionのnormativeな参照
  - **W3C Member Submissionは標準化されたものではない**
  - WS-SX TCのチャータには、標準化プロセスに乗っていない仕様、標準化プロセスが十分進んでいない仕様は記述を抽象化すると明記
  - (Web Services Policy 1.5は2007年2月28日にW3C Candidate Recommendation)
- WS-TrustはそのままOASIS Standardとなったが、WS-Policyを参照した仕様をもつ次のTCも巻き込んだ議論に発展
  - WS-TX TC(3月にOASISメンバー投票)
  - WS-RX TC(3月時点でTC内での最終段階、6月にOASISメンバー投票)
- 4月3日、BEA、富士通、日立、IBM、Microsoft、Nokia、Oracle、SAPの連名で、WS-Policy 1.5の完成後直ちにWS-RX/SX/TXの仕様の参照を同期して改版することを提案
  →WS-TX TC、WS-SX TCで、そのためのチャータ変更が決定
- 今後、WS-Policy完成後適切に仕様が修正されることを確実にするとともに、それまでは仕様の利用に注意が必要

WSFED TCチャータの問題

- OASISではTC設立時にチャータ案へのコメントを受け付ける期間を設定
  - OASIS事務局とTC召集者(Convener)でコメントに関して電話会議を開催
  - 提案者グループは各コメントへの対処を公表する義務
- WSFED TCのチャータ案に対して7社9人から31件のコメント
  - WS-SX/RX/TXで問題となったことが起こらないよう、非標準仕様の参照に関してチャータの明確化を要求するコメント
    - W3Cに提出されたが**標準化作業が開始していない仕様**
      - WS-Transfer
      - WS-ResourceTransfer
      - WS-Eventing
    - **標準化団体に提出されていない仕様**
      - WS-MetadataExchange

    } **注意が必要**
  - SAMLなどとの機能的重なりを指摘するコメント
- 4月5日の電話会議でのTC招集者からの回答
  - 単純ミスの指摘以外全て「**No changes are required**」
- OASISのプロセスでは、コメントは出せるが、提案者がそれに対して一方的に回答するだけよい
- OASISは規格の収束を奨励するが義務化しない
- WSFED TCのチャータには問題点が残ったままであり、今後注意が必要

まとめ

- 2002年のWS-SecurityのOASIS提出からWebサービスのセキュリティ規格の標準化開始
- 他の規定の標準化はなかなか始まらなかったが、2005年末頃から標準化の動きは早くなってきた
- 多くの仕様が関係しあい、その標準化の進行がバラバラであることによる問題が表面化→**非標準仕様の参照**
  - 標準規格を実装するためには、非標準の仕様を実装することが必要!?
  - **IPR(知的財産権)**の問題
- WS-Federation標準化開始により、今後のSAML(Liberty)との関係に要注目

- 参考
  - WS-Policy 1.5の完成後直ちにWS-RX/SX/TXの仕様の参照を同期して改版することの提案
    http://lists.oasis-open.org/archives/ws-sx/200704/msg00000.html
  - WSFED TCチャータ案に対するコメント
    http://lists.oasis-open.org/archives/oasis-charter-discuss/
    (上記URLの中にある200703および200704)
  - コメントに対する回答
    http://www.oasis-open.org/committees/documents.php?wg_abbrev=oasis-charter-discuss
    (上記URLの中にあるOASIS WSFED Comment Log Apr-09-07.pdf)

仕様解説

WS-Securityの機能

- SOAPメッセージに対する基本的セキュリティ機能を提供
  - メッセージ完全性
    - SOAPメッセージが送信の途中で改変されないこと
  - メッセージ秘匿性
    - SOAPメッセージが送信の途中で他者に漏洩しないこと
  - メッセージ送信者認証
    - SOAPメッセージの送信者の認証
- これらの機能を提供するために次の手段を規定
  - SOAPメッセージにおけるXML Signatureの利用方法
  - SOAPメッセージにおけるXML Encryptionの利用方法
  - SOAPメッセージにおいて各種セキュリティ・トークンの利用方法
    - ユーザ名、X.509 証明書、SAMLアサーションなど個別のプロファイルを規定

Securityヘッダ

- セキュリティ関連情報は<wsse:Security>ヘッダ･ブロックに記述

```
<S:Envelope>
  <S:Header>
    <wsse:Security>
      署名関連情報
      暗号化関連情報
      セキュリティトークン
      タイムスタンプ
    </wsse:Security>
  </S:Header>
  <S:Body>
  ...
  </S:Body>
</S:Envelope>
```


(参考)署名

- XML Signatureの<ds:Signature>を<wsse:Security>ヘッダ･ブロックで利用

```
<S:Envelope>
  <S:Header>
    <wsse:Security>
      <wsse:BinarySecurityToken ...>
        ...
      </wsse:BinarySecurityToken>
      <ds:Signature>
          鍵情報(セキュリティ･トークン)への参照
          署名対象への参照
      </ds:Signature>
    </wsse:Security>
  </S:Header>
  <S:Body wsu:Id="body">
  ...
  </S:Body>
</S:Envelope>
```

署名情報

(参考)暗号化(1)

- XML Encryptionの<xenc:ReferenceList>を<wsse:Security>ヘッダ・ブロックで利用
- 暗号化されたデータは<xenc:EncryptedData>で表される

```
<S:Envelope>
  <S:Header>
    <wsse:Security>
      <xenc:ReferenceList>
        <xenc:DataReference URI=... />
      </xenc:ReferenceList>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <xenc:EncryptedData Id="...">
      ...
    </xenc:EncryptedData>
  </S:Body>
</S:Envelope>
```

暗号化された場所のリスト

暗号化されたデータへの参照

暗号化されたデータ

鍵情報(への参照)も

(参考)暗号化(2)

- XML Encryptionの<xenc:EncryptedKey>を<wsse:Security>で利用
- 暗号化されたデータは<xenc:EncryptedData>で表される

```
<S:Envelope>
  <S:Header>
    <wsse:Security>
      <xenc:EncryptedKey>
        ...
        <xenc:ReferenceList>
          ...
        <xenc:ReferenceList>
      </xenc:EncryptedKey>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <xenc:EncryptedData Id="...">
      ...
    </xenc:EncryptedData>
  </S:Body>
</S:Envelope>
```

暗号化に利用した対称鍵

鍵情報(への参照)

暗号化されたデータへの参照

暗号化されたデータ

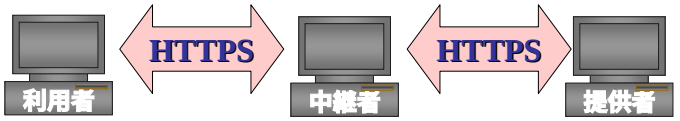
SSL/TLSとWS-Securityの比較

技術	SSL/TLS	WS-Securityとそのプロファイル
レイヤー	トランスポート層のセキュリティ	メッセージ層のセキュリティ **トランスポート層に依存しない**
完全性	HTTPヘッダ、ボディを含むメッセージ全体の完全性のみ HTTPセッションの間のみ有効で、SOAPメッセージが受信されてからの保護はない 中継者がいる場合は不適	SOAPメッセージの部分的な完全性を提供可能 中継者がいる場合に有効 複数SOAP受信者に対しても完全性を提供可能
秘匿性	メッセージ全体の秘匿性のみ HTTPセッションの間のみ有効で、SOAPメッセージが受信されてからの保護はない 中継者がいる場合は不適	メッセージの部分への秘匿性を提供可能 中継者がいる場合に有効
認証	サーバ証明書によりHTTPサービスを認証 クライアント証明書によりHTTPユーザ・エージェントの認証が可能 セッションに対する認証	SOAP Sender (メッセージの元々の送信者や中継者)の認証 **さまざまなトークンを利用可能(ユーザ名、証明書、SAMLなど)**
大きな違い	**ポイント・ツー・ポイントでのセキュリティ** **セッションの間のみ有効なセキュリティ**	**エンド・ツー・エンドでのセキュリティ** **セキュアな形でメッセージ保存可能** **→監査ログとしても有効に利用可能**

実際には両方を組み合わせて利用

エンドツーエンドのセキュリティ

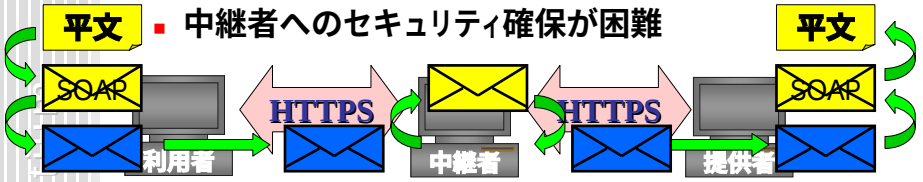
- SSL: Point-to-Point
  - **トランスポート層のセキュリティ**
  - **中継者へのセキュリティ確保が困難**



- WSS: End-to-End
  - **メッセージコンテントのセキュリティ**
  - **中継者にもセキュリティを確保**

エンドツーエンドのセキュリティ

- SSL: Point-to-Point
  - トランスポート層のセキュリティ
  - 中継者へのセキュリティ確保が困難



- WSS: End-to-End
  - メッセージコンテントのセキュリティ
  - 中継者にもセキュリティを確保

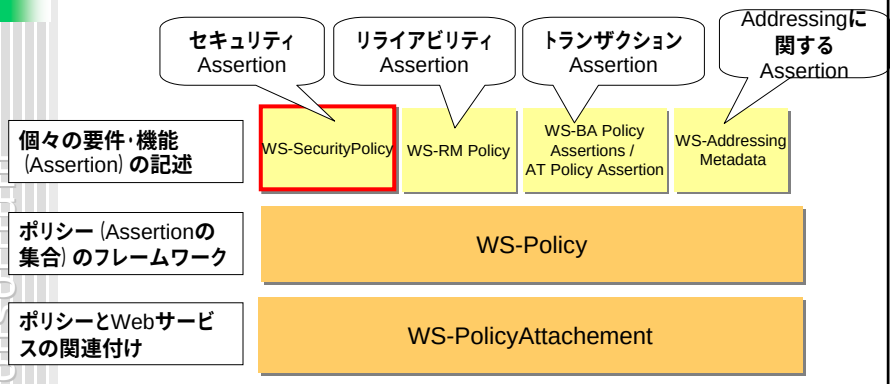
Webサービスポリシーの概要

- 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



WS-Policy、WS-PolicyAttachment:
　W3Cで標準化中の仕様の正式名称はそれぞれ
　・Web Services Policy 1.5 – Framework
　・Web Services Policy 1.5 – Attachment
WS-RM：WS-ReliableMessaging
WS-BA：WS-BusinessActivity
WS-AT：WS-AtomicTransaction
(トランザクションに関するAssertionは個別仕様書ではなく本仕様の中で規定)

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-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-SecurityPolicyが規定するアサーション

- Protection Assertions
  - メッセージ保護に関するアサーション (どのような保護が必要か)
- Token Assertions
  - メッセージで利用されるトークンを指定するアサーション (どのトークンを使うか)
- Security Binding Assertions
  - セキュリティがどのような機構(トランスポート層、メッセージ層)で提供されるかを指定するアサーション (保護のためにどの技術を使うか)
- Supporting Tokens Assertions
  - サービスが複数クレームを要求するような場合に、メッセージに含めるべき追加のトークンを指定するアサーション
- Protocol Assertions
  - WSS (1.0, 1.1)、WS-Trustのオプションを指定するアサーション

Protection Assertions

- 完全性(署名)に関するアサーション(Integrity Assertions)
  - SignedParts
    - ボディ全体、個々のヘッダ、添付全体を署名すべき部分として指定可能
  - SignedElements
    - XPathを使用して個別の要素単位で署名すべき部分を指定可能
- 秘匿性(暗号化)に関するアサーション(Confidentiality Assertions)
  - EncryptedParts
    - ボディ全体、個々のヘッダ、添付全体を暗号化すべき部分として指定可能
  - EncryptedElements
    - XPathを使用して個別の要素単位で暗号化すべき部分を指定可能(#Element暗号)
  - ContentEncryptedElements
    - XPathを使用して個別の要素単位で暗号化すべき部分を指定可能(#Content暗号)

```
<sp:SignedParts>
  <sp:Body/>
  <sp:Header Name="To" Namespace=".../addressing"/>
</sp:SignedParts>
```

SOAPボディとToヘッダの完全性が確保されなければならない

```
<sp:EncryptedElements>
  <sp:XPath>S:body/Order/CardInfo</sp:XPath>
</sp:EncryptedElements>
```

SOAPボディの下のCardInfo要素の秘匿性が確保されなければならない

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:definitions>
  <wsp:Policy wsu:Id="ABC">
    ...
  </wsp:Policy>
  <wsp:Policy wsu:Id="XYZ">
    ...
  </wsp:policy>
  <wsdl:binding name="…" type="…">
    <wsp:PolicyReference URI="#ABC" wsdl:required="true" />
    <wsdl:operation …>
      <wsdl:input>
        <wsp:PolicyReference URI="#XYZ" wsdl:required="true" />
        ...
      </wsdl:input>
      ...
</wsdl:definition>
```

様々なWSDL要素に対してポリシーを設定可能

ポリシー

エンドポイントに対するポリシー

ポリシー

入力メッセージに対するポリシー

WSDLの中に埋め込む場合は、<wsdl:definitions>の直下にポリシーを置き、wsp:PolicyReferenceで参照することを推奨