HITACHI
Inspire the Next

Cosminexus
Web サービスセキュリティ 使用の手引

解説・手引・文法書

3020-3-M48-20

マニュアルの購入方法

このマニュアル，および関連するマニュアルをご購入の際は，巻末の「ソフトウェアマニュアルのサービス ご案内」をご参照ください。

■対象製品

●**適用** OS：Windows Server 2003，Windows Server 2003 R2，Windows Vista，Windows XP

P-2443-7F74 uCosminexus Developer Professional 07-60

P-2443-7T74 uCosminexus Service Architect 07-60 ※

●**適用** OS：Windows Server 2003，Windows Server 2003 R2，Windows Server 2003（x64），Windows Server 2003 R2（x64）

P-2443-7K74 uCosminexus Application Server Enterprise 07-60

P-2443-7S74 uCosminexus Service Platform 07-60 ※

●**適用** OS：AIX 5L V5.2，AIX 5L V5.3

P-1M43-7K71 uCosminexus Application Server Enterprise 07-60 ※

P-1M43-7S71 uCosminexus Service Platform 07-60 ※

●**適用** OS：HP-UX 11i V2（IPF），HP-UX 11i V3（IPF）

P-1J43-7K71 uCosminexus Application Server Enterprise 07-60 ※

●**適用** OS：Red Hat Enterprise Linux AS 3（x86），Red Hat Enterprise Linux AS 4（x86），Red Hat Enterprise Linux ES 3（x86），Red Hat Enterprise Linux ES 4（x86），Red Hat Enterprise Linux AS 3（AMD64 & Intel EM64T），Red Hat Enterprise Linux AS 4（AMD64 & Intel EM64T），Red Hat Enterprise Linux ES 3（AMD64 & Intel EM64T），Red Hat Enterprise Linux ES 4（AMD64 & Intel EM64T）

P-9S43-7K71 uCosminexus Application Server Enterprise 07-60 ※

●**適用** OS：Red Hat Enterprise Linux AS 3（x86），Red Hat Enterprise Linux AS 4（x86），Red Hat Enterprise Linux 5 Advanced Platform（x86），Red Hat Enterprise Linux ES 3（x86），Red Hat Enterprise Linux ES 4（x86），Red Hat Enterprise Linux 5（x86），Red Hat Enterprise Linux AS 3（AMD64 & Intel EM64T），Red Hat Enterprise Linux AS 4（AMD64 & Intel EM64T），Red Hat Enterprise Linux 5 Advanced Platform（AMD/Intel 64），Red Hat Enterprise Linux ES 3（AMD64 & Intel EM64T），Red Hat Enterprise Linux ES 4（AMD64 & Intel EM64T），Red Hat Enterprise Linux 5（AMD/Intel 64）

P-9S43-7S71 uCosminexus Service Platform 07-60 ※

●**適用** OS：Red Hat Enterprise Linux AS 3（IPF），Red Hat Enterprise Linux AS 4（IPF），Red Hat Enterprise Linux 5 Advanced Platform（Intel Itanium）

P-9V43-7K71 uCosminexus Application Server Enterprise 07-60 ※

●**適用** OS：Solaris 9，Solaris 10

P-9D43-7K71 uCosminexus Application Server Enterprise 07-60 ※

P-9D43-7S71 uCosminexus Service Platform 07-60 ※

※印の製品については，サポート時期をご確認ください。

上記のプログラムプロダクトのほかにもこのマニュアルをご利用になれる場合があります。詳細は「リリースノート」でご確認ください。

本製品では日立トレース共通ライブラリをインストールします。

■輸出時の注意

本製品を輸出される場合には，外国為替および外国貿易法ならびに米国の輸出管理関連法規などの規制をご確認の上，必要な手続きをお取りください。

なお，ご不明な場合は，弊社担当営業にお問い合わせください。

■商標類

AIX は，米国における米国 International Business Machines Corp. の登録商標です。
AMD, AMD Opteron およびその組み合わせは，Advanced Micro Devices, Inc. の商標です。
HP-UX は，米国 Hewlett-Packard Company のオペレーティングシステムの名称です。
Intel は，Intel Corporation の会社名です。
Itanium は，アメリカ合衆国および他の国におけるインテル コーポレーションまたはその子会社の登録商標です。
Java 及びすべての Java 関連の商標及びロゴは，米国及びその他の国における米国 Sun Microsystems,Inc. の商標または登録商標です。
JDK は，米国 Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。
Linux は，Linus Torvalds の米国およびその他の国における登録商標あるいは商標です。
Microsoft は，米国およびその他の国における米国 Microsoft Corp. の登録商標です。
Red Hat は，米国およびその他の国で Red Hat, Inc. の登録商標若しくは商標です。
SOAP（Simple Object Access Protocol）は，分散ネットワーク環境において XML ベースの情報を交換するための通信プロトコルの名称です。
Solaris は，米国 Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。
Sun，Sun Microsystems，Java は，米国 Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。
UNIX は，X/Open Company Limited が独占的にライセンスしている米国ならびに他の国における登録商標です。
Windows は，米国およびその他の国における米国 Microsoft Corp. の登録商標です。
Windows Server は，米国 Microsoft Corporation の米国及びその他の国における登録商標です。
Windows Vista は，米国 Microsoft Corporation の米国及びその他の国における登録商標です。
This product includes software developed by the Apache Software Foundation (http://www.apache.org/).
プログラムプロダクト「P-9D43-7K71，P-9D43-7S71」には，米国 Sun Microsystems,Inc. が著作権を有している部分が含まれています。
プログラムプロダクト「P-9D43-7K71，P-9D43-7S71」には，UNIX System Laboratories,Inc. が著作権を有している部分が含まれています。

■発行

2006 年 4 月（第 1 版）3020-3-M48
2007 年 12 月（第 2 版）3020-3-M48-20

■著作権

変更内容

変更内容（3020-3-M48-20）uCosminexus Developer Professional 07-60，uCosminexus Application Server Enterprise 07-60，uCosminexus Service Architect 07-60，uCosminexus Service Platform 07-60

追加・変更内容	変更個所
次の前提 OS を削除した。 • Windows 2000 Server • Windows 2000 Professional	2.1.1
次の前提 OS を追加した。 • Windows Vista	2.1.1
次の前提 OS を削除した。 • AIX 5L V5.1 • Solaris 8	2.2.1
次の前提 OS を追加した。 • HP-UX 11i V3（IPF） • Red Hat Enterprise Linux ES 3（AMD64 & Intel EM64T） • Red Hat Enterprise Linux ES 4（AMD64 & Intel EM64T） • Red Hat Enterprise Linux 5 Advanced Platform（x86） • Red Hat Enterprise Linux 5（x86） • Red Hat Enterprise Linux 5 Advanced Platform（AMD/Intel 64） • Red Hat Enterprise Linux 5（AMD/Intel 64） • Red Hat Enterprise Linux AS 3（IPF） • Red Hat Enterprise Linux AS 4（IPF） • Red Hat Enterprise Linux 5 Advanced Platform（Intel Itanium）	2.2.1
WS-Security v1.1 対応に伴い，次の内容を変更した。 • WSSUsernameToken.PasswordType クラスの列挙値の形式 • WSSUsernameToken.PasswordType インタフェース（PasswordType 要素の操作）のクラス定義，およびパスワード種別の形式 • Fault コードの接頭辞，および名前空間 • WS-Security 標準仕様 • WS-Security 仕様のサポート範囲 • Web サービスセキュリティ機能定義ファイルの項目（BinarySecurityTokenConfig，KeyIdentifier，Password） • Web サービスセキュリティポリシー定義ファイルの項目（TokenValidation）	表 5-16，5.5，7.2.3，7.2.9，付録 A，表 A-1，表 C-12，表 C-24，表 C-35，表 C-62
メッセージを変更した。 KDCGF0001-E ～ KDCGF0008-E，KDCGW0001-E ～ KDCGW0003-E，KDCGW9000-E	7.2.3，7.2.9
メッセージを追加した。 KDCGF0009-E，KDCGF0010-E	7.2.3
下位バージョンからの移行手順を追加した。	付録 B

単なる誤字・脱字などはお断りなく訂正しました。

はじめに

このマニュアルは，Cosminexus が提供する Cosminexus Web Services - Security の Web サービスセキュリティ機能について説明したものです。
Cosminexus Web Services - Security は，Cosminexus を構成する次のプログラムプロダクトで提供されています。

- P-2443-7F74 uCosminexus Developer Professional
- P-2443-7T74 uCosminexus Service Architect
- P-2443-7K74 uCosminexus Application Server Enterprise
- P-2443-7S74 uCosminexus Service Platform
- P-1M43-7K71 uCosminexus Application Server Enterprise
- P-1M43-7S71 uCosminexus Service Platform
- P-1J43-7K71 uCosminexus Application Server Enterprise
- P-9S43-7K71 uCosminexus Application Server Enterprise
- P-9S43-7S71 uCosminexus Service Platform
- P-9V43-7K71 uCosminexus Application Server Enterprise
- P-9D43-7K71 uCosminexus Application Server Enterprise
- P-9D43-7S71 uCosminexus Service Platform

Cosminexus Web Services - Security の XML 署名・暗号処理機能については，マニュアル「Cosminexus XML Security - Core　ユーザーズガイド」を参照してください。

■対象読者

このマニュアルは，Cosminexus が提供する SOAP アプリケーション開発支援機能を利用して開発した SOAP アプリケーションに対して，Web サービスセキュリティ機能を使用する方を対象としています。また，このマニュアルをご利用になる方は，XML，SOAP，およびセキュリティに関する基本的な事項を理解されていることを前提としています。

■マニュアルの構成

このマニュアルは，次に示す章と付録から構成されています。

第 1 章　Web サービスセキュリティの概要

Web サービスセキュリティとは何か，また Cosminexus が提供する Web サービスセキュリティの機能について説明しています。

第 2 章　開発または実行に必要な製品

Web サービスセキュリティ機能を使用するために必要な OS，およびプログラムプロダクトについて説明しています。

第 3 章　Web サービスセキュリティ機能を使用する

Web サービスセキュリティ機能を使用する場合に必要な設定，および実装について説明していま

す。

第 4 章　Web サービスセキュリティ機能が提供するコマンド
Web サービスセキュリティ機能が提供するコマンドの形式やオプションなどを説明しています。

第 5 章　Web サービスセキュリティ機能が提供する API
Web サービスセキュリティ機能が提供する API の構文や引数など，API の仕様について説明しています。

第 6 章　障害対策
障害が発生した場合の対処方法を説明しています。

第 7 章　メッセージ一覧
Web サービスセキュリティ機能が出力するメッセージの内容および対処方法などについて説明しています。

付録 A　標準仕様への対応
Web サービスセキュリティ機能がサポートする WS-Security 標準仕様，XML 署名標準仕様，および XML 暗号標準仕様について説明しています。

付録 B　下位バージョンからの移行手順
Web サービスセキュリティ機能の下位バージョンからの移行手順について説明しています。

付録 C　定義ファイルの項目の詳細
Web サービスセキュリティ機能を使用する場合に必要な定義ファイルの要素名および指定回数を説明しています。また，各項目の役割についても説明しています。

付録 D　用語解説
このマニュアルで使用している用語の意味を説明しています。

■関連マニュアル

このマニュアルをご利用するに当たって，必要に応じて次に示すマニュアルを参照してください。

- Cosminexus　機能解説（3020-3-M03）
- Cosminexus　リファレンス　定義編（3020-3-M11）
- Cosminexus　SOAP アプリケーション開発ガイド（3020-3-M47）
- Cosminexus XML Security - Core　ユーザーズガイド（3020-3-M49）
- Cosminexus XML Processor　ユーザーズガイド（3020-3-M44）

■読書手順

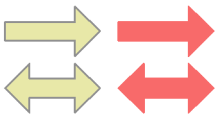
このマニュアルをご利用になるときは，目的に応じて必要な章をお読みください。各章の利用目的の例を次に示しますので，ご利用の際の目安にしてください。

章タイトル	利用目的の例
第 1 章　Web サービスセキュリティの概要	Web サービスセキュリティとは何かを知りたい。 Web サービスセキュリティ機能とは何かを知りたい。
第 2 章　開発または実行に必要な製品	Web サービスセキュリティ機能を使用するために必要な OS とソフトウェアの種類およびバージョンを知りたい。
第 3 章　Web サービスセキュリティ機能を使用する	Web サービスセキュリティ機能を使用するための設定方法を知りたい。 Web サービスセキュリティ機能の実装方法を知りたい。
第 4 章　Web サービスセキュリティ機能が提供するコマンド	どのようなコマンドがあるのかを知りたい。 コマンドの使用方法を知りたい。
第 5 章　Web サービスセキュリティ機能が提供する API	どのような API があるのかを知りたい。 API の使用方法を知りたい。
第 6 章　障害対策	障害が発生した場合の対処方法を知りたい。 トレースファイルの収集方法を知りたい。 アプリケーションログの収集方法を知りたい。
第 7 章　メッセージ一覧	メッセージが出力された場合の要因や対処方法を知りたい。
付録 A　標準仕様への対応	WS-Security 標準仕様のサポート範囲を知りたい。 XML 署名標準仕様のサポート範囲を知りたい。 XML 暗号標準仕様のサポート範囲を知りたい。
付録 B　下位バージョンからの移行手順	下位バージョンからバージョンアップする場合の移行方法を知りたい。
付録 C　定義ファイルの項目の詳細	定義ファイルで設定する項目の要素名，説明，または指定回数を知りたい。
付録 D　用語解説	このマニュアルで使用されている用語の意味を知りたい。

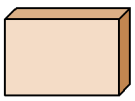
■図中で使用する記号

このマニュアルの図中で使用する記号を，次のように定義します。

●データの流れ



●プログラム



●工程，作業項目の流れ

■このマニュアルで使用している記号

このマニュアルで使用している記号を次のように定義します。

記号	意味
< >	< > で囲まれた部分は状況に応じて変化する内容であることを示します。

■このマニュアルでの表記

このマニュアルでは，製品名を次のように表記しています。

製品名			略称
IPF			Itanium(R) Processor Family
UNIX	AIX		AIX 5L V5.2 AIX 5L V5.3
	HP-UX	HP-UX(IPF)	HP-UX 11i V2.0（IPF） HP-UX 11i V3.0（IPF）
	Linux	Linux(x86, AMD64 & Intel EM64T)	Red Hat Enterprise Linux AS 3（x86） Red Hat Enterprise Linux AS 4（x86） Red Hat Enterprise Linux 5 Advanced Platform（x86） Red Hat Enterprise Linux ES 3（x86） Red Hat Enterprise Linux ES 4（x86） Red Hat Enterprise Linux 5（x86） Red Hat Enterprise Linux AS 3（AMD64 & Intel EM64T） Red Hat Enterprise Linux AS 4（AMD64 & Intel EM64T） Red Hat Enterprise Linux 5 Advanced Platform（AMD/Intel 64） Red Hat Enterprise Linux ES 3（AMD64 & Intel EM64T） Red Hat Enterprise Linux ES 4（AMD64 & Intel EM64T） Red Hat Enterprise Linux 5（AMD/Intel 64）
		Linux(IPF)	Red Hat Enterprise Linux AS 3（IPF） Red Hat Enterprise Linux AS 4（IPF） Red Hat Enterprise Linux 5 Advanced Platform（Intel Itanium）
	Solaris		Solaris 9 Solaris 10
Windows	Windows Server 2003		Microsoft(R) Windows Server(R) 2003, Standard Edition Operating System Microsoft(R) Windows Server(R) 2003, Enterprise Edition Operating System
	Windows Server 2003(x64)		Microsoft(R) Windows Server(R) 2003, Standard x64 Edition Operating System Microsoft(R) Windows Server(R) 2003, Enterprise x64 Edition Operating System

製品名		略称
	Windows Server 2003 R2	Microsoft(R) Windows Server(R) 2003 Release 2, Standard Edition Operating System Microsoft(R) Windows Server(R) 2003 Release 2, Enterprise Edition Operating System
	Windows Server 2003 R2(x64)	Microsoft(R) Windows Server(R) 2003 Release 2, Standard x64 Edition Operating System Microsoft(R) Windows Server(R) 2003 Release 2, Enterprise x64 Edition Operating System
	Windows Vista	Microsoft(R) Windows Vista(R) Business
		Microsoft(R) Windows Vista(R) Enterprise
		Microsoft(R) Windows Vista(R) Ultimate
	Windows XP	Microsoft(R) Windows(R) XP Professional Operating System

■このマニュアルで使用する略語

このマニュアルでは，次に示す略語を使用しています。

略語	正式名称
API	Application Programming Interface
EJB	Enterprise Java Beans
J2SE	Java 2 Platform, Standard Edition
JAAS	Java Authentication and Authorization Service
JAR	Java Archive
OASIS	Organization for the Advancement of Structured Information Standards
OS	Operating System
RPC	Remote Procedure Call
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WAR	Web Archive
WS	Web Service
XML	Extensible Markup Language

■このマニュアルで使用する名前空間

このマニュアルでは，次に示す名前空間を使用しています。

プレフィックス	名前空間
ds	http://www.w3.org/2000/09/xmldsig#

プレフィックス	名前空間
soapenv	http://www.w3.org/2001/12/soap-envelope
wsse	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
wsu	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd

■適用 OS の違いによる機能相違点の表記

OS によって記述を書き分ける場合，次に示す表記を使用しています。

表記	意味
Windows の場合	Windows 系の OS を使用している場合
その他の OS の場合	Windows 系以外の OS（HP-UX，AIX，Linux，または Solaris）を使用している場合

■ Windows の場合のフォルダとパスの表記

このマニュアルでは，Windows，HP-UX，AIX，Linux および Solaris で共通の内容の場合，Windows の「フォルダ」を「ディレクトリ」と表記しています。また，「¥」を「/」と表記しています。

Windows の場合，「ディレクトリ」を「フォルダ」に，「/」を「¥」に置き換えてお読みください。

■常用漢字以外の漢字の使用について

このマニュアルでは，常用漢字を使用することを基本としていますが，次に示す用語については，常用漢字以外の漢字を使用しています。

鍵（かぎ） 個所（かしょ） 必須（ひっす） 漏洩（ろうえい）

■ KB（キロバイト）などの単位表記について

1KB（キロバイト），1MB（メガバイト），1GB（ギガバイト），1TB（テラバイト）はそれぞれ 1,024 バイト，$1{,}024^2$ バイト，$1{,}024^3$ バイト，$1{,}024^4$ バイトです。

目次

1 Web サービスセキュリティの概要

この章では，Web サービスセキュリティとは何か，また，Web サービスセキュリティと SOAP，および XML セキュリティとの関係について説明します。また，Cosminexus が提供する Web サービスセキュリティ機能についても説明します。

1.1 Web サービスセキュリティとは

Web サービスセキュリティとは，広義には Web サービスを安全に実現するためのセキュリティ技術全般を指します。現在，代表的な Web サービスセキュリティは，XML 署名および XML 暗号を利用した XML セキュリティです。また，Web サービスの実現には，SOAP を使用するのが一般的です。この節では，Web サービスセキュリティと SOAP，および XML セキュリティとの関係について説明します。

1.1.1 Web サービスセキュリティと SOAP との関係

Web サービスの多くは，SOAP メッセージを送受信することで実現します。SOAP メッセージは XML 形式のデータです。そのため，ネットワークを経由して，そのまま SOAP メッセージを送信した場合，データの内容を改ざんされたり，カード番号などの重要な情報を第三者に盗聴されたりしてしまうおそれがあります。このような改ざんや盗聴を防止するための技術が Web サービスセキュリティです。

Cosminexus Web Services - Security は，Cosminexus Component Container の SOAP 通信基盤機能に組み込んで，Web サービスセキュリティに対応した SOAP メッセージを送受信する場合に使用します。Cosminexus Web Services - Security は，SOAP 通信基盤機能の互換／標準モードのどちらでも使用できます。

また，Cosminexus Component Container の SOAP アプリケーション開発支援機能で Web サービスセキュリティ対応の SOAP アプリケーションを開発する場合にも，Cosminexus Web Services - Security を使用します。SOAP アプリケーションの開発については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

1.1.2 Web サービスセキュリティと XML セキュリティとの関係

W3C が規定している XML 署名および XML 暗号の仕様を XML セキュリティと呼びます。XML セキュリティを利用すると，SOAP メッセージに XML 署名を付与したり，SOAP メッセージを暗号化したりできます。代表的な Web サービスセキュリティは，OASIS で規定されているもので，XML セキュリティを利用した安全な Web サービスを実現します。

Cosminexus Web Services - Security は，XML 署名・暗号処理機能を利用した Web サービスセキュリティ機能を提供します。XML 署名・暗号処理機能については，マニュアル「Cosminexus XML Security - Core　ユーザーズガイド」を参照してください。

1.2 Cosminexus が提供する Web サービスセキュリティ機能

Cosminexus では，Cosminexus Web Services - Security を利用して Web サービスセキュリティを実現します。Cosminexus Web Services - Security が提供する機能のうち，SOAP メッセージの完全性や秘匿性などのセキュリティを提供する機能を Web サービスセキュリティ機能と呼びます。

1.2.1 SOAP メッセージの完全性を保証する

Web サービスセキュリティ機能は，SOAP メッセージの完全性を保証します。SOAP メッセージが送受信中に改ざんされないよう，SOAP メッセージの完全性を保証するための方法として，署名を利用する方法があります。

Web サービスセキュリティ機能を使用すると，SOAP メッセージに署名を付与できます。SOAP メッセージに署名が付与されている場合，SOAP メッセージを受信したときに署名を検証することによって，SOAP メッセージが送受信中に改ざんされていないかどうかを調べられます。また，SOAP メッセージに証明書が付与されている場合は，Web サービスセキュリティ機能で証明書も検証できます。

1.2.2 SOAP メッセージの秘匿性を保証する

Web サービスセキュリティ機能は，SOAP メッセージの秘匿性を保証します。SOAP メッセージが送受信中に第三者によって盗聴されないよう，SOAP メッセージの秘匿性を保証するための方法として，SOAP メッセージを暗号化する方法があります。

Web サービスセキュリティ機能を使用すると，必要な部分だけを指定して SOAP メッセージを暗号化することができます。Web サービスセキュリティ機能では，暗号化に XML 暗号を使用します。暗号化することによって，SOAP メッセージの送受信中に第三者にメッセージの内容を盗聴されるおそれがなくなります。

1.2.3 SOAP メッセージの認証をサポート

Web サービスセキュリティ機能は，SOAP メッセージの認証および証明書の検証をサポートしています。SOAP メッセージの送信者を特定する必要がある場合は，ユーザー名やパスワードを SOAP メッセージに含めるように設定できます。

2 開発または実行に必要な製品

この章では，Web サービスセキュリティ機能の前提 OS および前提プログラムを，開発時と実行時に分けて説明します。また，開発時と実行時のプログラム構成例をそれぞれ紹介します。

2.1　開発に必要な製品

この節では，Cosminexus Web Services - Security が提供する Web サービスセキュリティ機能を SOAP アプリケーション開発支援機能に組み込んで SOAP アプリケーションを開発する場合の，前提 OS および前提プログラムについて説明します。また，開発時のプログラム構成例についても説明します。

2.1.1　開発時の前提 OS

Web サービスセキュリティ機能を SOAP アプリケーション開発支援機能に組み込んで開発する場合の前提 OS は，次のどれかです。

- Windows Vista
- Windows XP
- Windows Server 2003
- Windows Server 2003 R2

2.1.2　開発時の前提プログラム

Web サービスセキュリティ機能を SOAP アプリケーション開発支援機能に組み込んで SOAP アプリケーションを開発する場合に，必要となる前提プログラムを次の表に示します。前提プログラムのバージョンについては「リリースノート」でご確認ください。

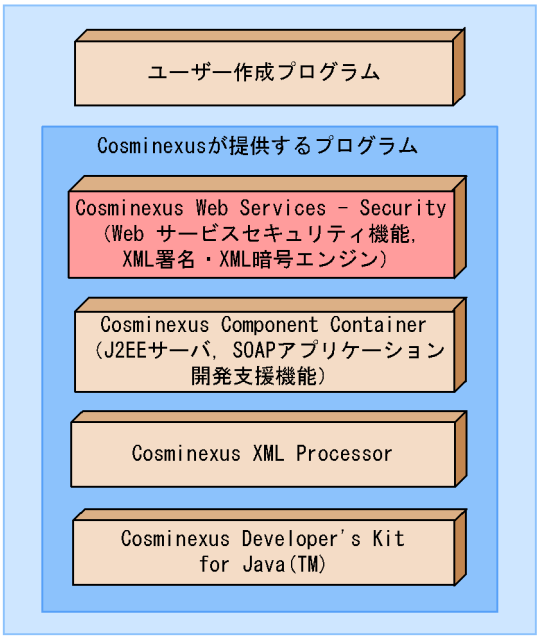
表 2-1　前提プログラム一覧（開発時）

分類	プログラム名
J2EE サーバ，SOAP アプリケーション開発支援機能	Cosminexus Component Container
Java™2SDK	Cosminexus Developer's Kit for Java™
XML プロセッサ	Cosminexus XML Processor

2.1.3　開発時のプログラム構成例

Web サービスセキュリティ機能を SOAP アプリケーション開発支援機能に組み込んで SOAP アプリケーションを開発する場合のプログラム構成例を次に示します。

図 2-1　Web サービスセキュリティ機能を組み込む場合のプログラム構成例



2.2　実行に必要な製品

この節では，Cosminexus Web Services - Security が提供する Web サービスセキュリティ機能を組み込んだ SOAP アプリケーションを実行する場合の，前提 OS および前提プログラムについて説明します。また，実行時のプログラム構成例についても説明します。

2.2.1　実行時の前提 OS

Web サービスセキュリティ機能を組み込んだ SOAP アプリケーションを実行する場合の前提 OS は，次のどれかです。

- Windows Server 2003
- Windows Server 2003（x64）
- Windows Server 2003 R2
- Windows Server 2003 R2（x64）
- Red Hat Enterprise Linux AS 3（x86）
- Red Hat Enterprise Linux AS 4（x86）
- Red Hat Enterprise Linux 5 Advanced Platform（x86）※ 1
- Red Hat Enterprise Linux ES 3（x86）
- Red Hat Enterprise Linux ES 4（x86）
- Red Hat Enterprise Linux 5（x86）※ 1
- Red Hat Enterprise Linux AS 3（AMD64 & Intel EM64T）
- Red Hat Enterprise Linux AS 4（AMD64 & Intel EM64T）
- Red Hat Enterprise Linux 5 Advanced Platform（AMD/Intel 64）※ 1
- Red Hat Enterprise Linux ES 3（AMD64 & Intel EM64T）
- Red Hat Enterprise Linux ES 4（AMD64 & Intel EM64T）
- Red Hat Enterprise Linux 5（AMD/Intel 64）※ 1
- Red Hat Enterprise Linux AS 3（IPF）
- Red Hat Enterprise Linux AS 4（IPF）
- Red Hat Enterprise Linux 5 Advanced Platform（Intel Itanium）※ 2
- Solaris 9
- Solaris 10
- AIX 5L V5.2
- AIX 5L V5.3
- HP-UX 11i V2（IPF）※ 2
- HP-UX 11i V3（IPF）※ 2

注※ 1　uCosminexus Service Platform だけに該当します。

注※ 2　uCosminexus Service Platform には該当しません。

2.2.2　実行時の前提プログラム

Web サービスセキュリティ機能を組み込んだ SOAP アプリケーションを実行する場合に必要となる前提プログラムを次の表に示します。前提プログラムのバージョンについては「リリースノート」でご確認ください。

表 2-2　前提プログラム一覧（実行時）

分類	プログラム名
Web サーバ	Windows の場合： Cosminexus Component Container が動作する Web サーバ※ その他の OS の場合： Hitachi Web Server
J2EE サーバ，SOAP 通信基盤（SOAP クライアントライブラリ，SOAP エンジン）	Cosminexus Component Container
Java™2SDK	Cosminexus Developer's Kit for Java™
XML プロセッサ	Cosminexus XML Processor

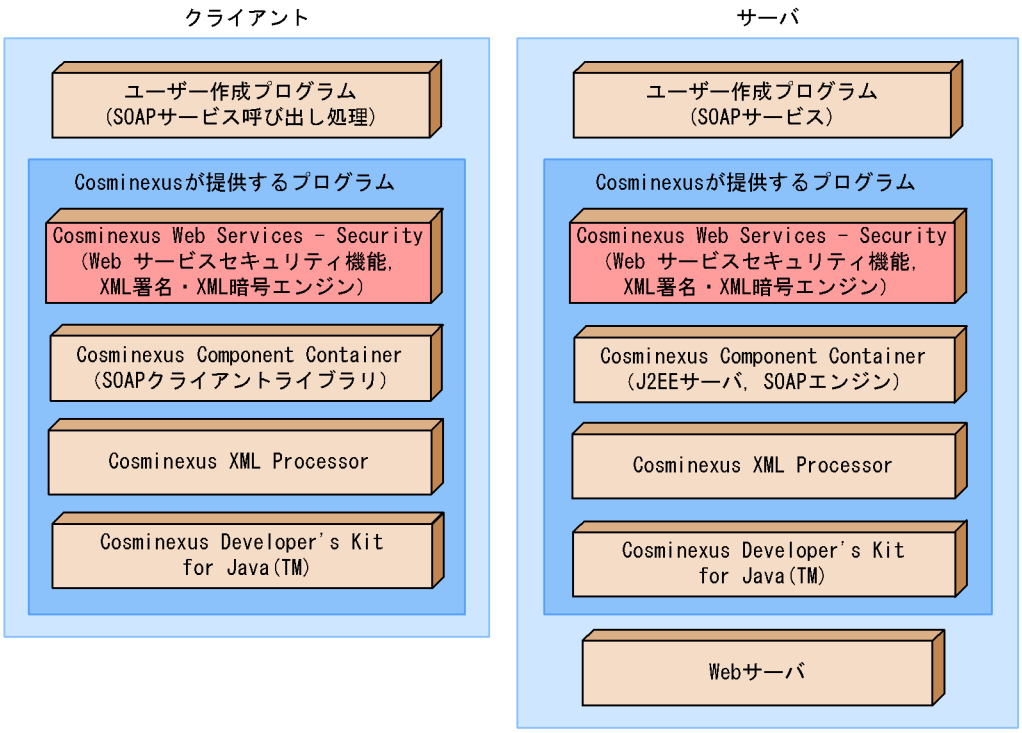
注※

Cosminexus Component Container が動作する Web サーバについての詳細は，マニュアル「Cosminexus　機能解説」を参照してください。

2.2.3　実行時のプログラム構成例

Web サービスセキュリティ機能を SOAP アプリケーション開発支援機能に組み込んで，SOAP アプリケーションを実行する場合の，クライアント側とサーバ側のプログラム構成例を次に示します。

図 2-2 Web サービスセキュリティ機能を組み込んだ SOAP アプリケーションを実行する場合のプログラム構成例



3 Web サービスセキュリティ機能を使用する

この章では，開発した SOAP アプリケーションに対して，Web サービスセキュリティ機能を使用する手順について説明します。

3.1 定義ファイルの設定

Web サービスセキュリティ機能を使用するためには，次に示す二つの定義ファイルに必要な項目を設定する必要があります。

- Web サービスセキュリティ機能定義ファイル（security-config.xml）
- Web サービスセキュリティポリシー定義ファイル（policy-config.xml）

これらの定義ファイルは，SOAP アプリケーションを利用するクライアント，およびサーバの両方に配置する必要があります。

次に，各定義ファイルの概要を説明します。

3.1.1 Web サービスセキュリティ機能定義ファイル

Web サービスセキュリティ機能定義ファイルは，Web サービスセキュリティ機能の動作を定義するためのファイルです。例えば，SOAP メッセージに署名を付与する場合，署名に用いる証明書を指定します。または，SOAP メッセージを暗号化する場合は，どの暗号化アルゴリズムを用いるのか，などを定義します。

Web サービスセキュリティ機能定義ファイルは，次の要素を持っています。

- BindingConfig
  Web サービスセキュリティの各機能で共通して使用する情報を設定します。
- RequestSenderConfig
  リクエストメッセージ送信時の Web サービスセキュリティ機能を設定します。
- ResponseSenderConfig
  レスポンスメッセージ送信時の Web サービスセキュリティ機能を設定します。
- RequestReceiverConfig
  リクエストメッセージ受信時の Web サービスセキュリティ機能を設定します。
- ResponseReceiverConfig
  レスポンスメッセージ受信時の Web サービスセキュリティ機能を設定します。

クライアントとサーバでは，必要な要素が異なります。

Web サービスセキュリティ機能を使用するために，クライアントおよびサーバで設定する必要がある要素を次の表に示します。

表 3-1 設定が必要な要素（Web サービスセキュリティ機能定義ファイル）

要素名	クライアント	サーバ
BindingConfig	○	○
RequestSenderConfig	○	×
ResponseSenderConfig	×	○

要素名	クライアント	サーバ
RequestReceiverConfig	×	○
ResponseReceiverConfig	○	×

（凡例）
○：必要
×：不要

Web サービスセキュリティ機能定義ファイルの項目については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

3.1.2　Web サービスセキュリティポリシー定義ファイル

Web サービスセキュリティポリシー定義ファイルは，受信した SOAP メッセージを検証するポリシーを定義するためのファイルです。Web サービスセキュリティ機能では，このポリシー定義ファイルを基に，SOAP メッセージが受信側の意図したものかどうかを検証します。

Web サービスセキュリティポリシー定義ファイルは，次の要素を持っています。

- GlobalConfig
  Web サービスセキュリティ機能のポリシー定義で共通して使用するポリシーを設定します
- RequestReceiverConfig
  リクエストメッセージ受信時のポリシーを設定します。
- ResponseReceiverConfig
  レスポンスメッセージ受信時のポリシーを設定します。

クライアントとサーバでは，必要な要素が異なります。

Web サービスセキュリティ機能を使用するために，クライアントおよびサーバで設定する必要がある要素を次の表に示します。

表 3-2　設定が必要な要素（Web サービスセキュリティポリシー定義ファイル）

要素名	クライアント	サーバ
GlobalConfig	○	○
RequestReceiverConfig	×	○
ResponseReceiverConfig	○	×

（凡例）
○：必要
×：不要

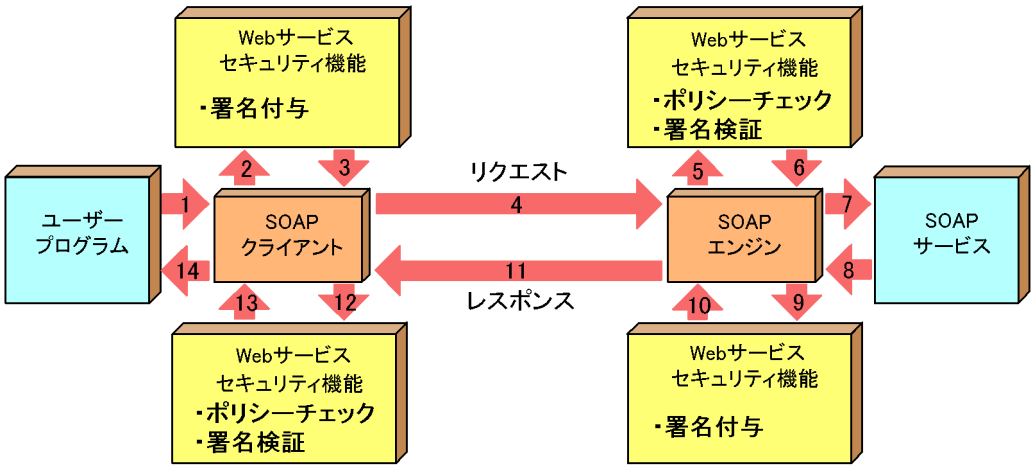
Web サービスセキュリティポリシー定義ファイルの項目については,「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

3.2　署名付与／検証機能を設定する

署名付与／検証機能は，SOAP 通信基盤で送信する SOAP メッセージに対して，XML 署名を付与したり，受信する SOAP メッセージに付与されている XML 署名を検証したりする機能です。署名付与機能は，SOAP メッセージの送信時に，署名検証機能は SOAP メッセージの受信時に使用します。

次に，署名付与／検証機能を使用した際のプログラムの構成図を示します。矢印およびその番号は，送受信するデータの処理の流れを示しています。

図 3-1　Web サービスセキュリティ機能使用時のプログラム構成（署名付与／検証）



署名付与／検証機能を使用する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
＜送信側＞
SecurityConfig
├── BindingConfig
└┬ RequestSenderConfig/ResponseSenderConfig
  └┬ SenderPortConfig
    └┬ RoleConfig
      ├── BinarySecurityTokenConfig
      └── SignatureConfig
＜受信側＞
SecurityConfig
├── BindingConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
  └┬ ReceiverPortConfig
    └── VerificationConfig
```

署名付与／検証機能を使用する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
  └┬ ReceiverPortConfig
    ├┬ SecurityTokenConfig
    │└── BinarySecurityTokenConfig
    └── VerificationConfig
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

Windows の場合，定義ファイルのサンプルは次のディレクトリに格納されていますので，これらを参考に定義ファイルを作成してください。なお，作成した定義ファイルの格納場所については，「3.9　Web サービスセキュリティ機能の実装手順」を参照してください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
signature/message/client/WEB-INF/classes
signature/message/service/WEB-INF/classes
signature/rpc/client/WEB-INF/classes
signature/rpc/service/WEB-INF/classes
```

また，Windows の場合，コーディングのサンプルは，次のディレクトリに格納されていますので，参考にしてください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
signature/KeyStore
signature/message/client/WEB-INF/classes/messagesampleclient
signature/message/service/WEB-INF/classes/messagesampleservice
signature/rpc/client/WEB-INF/classes/localhost
signature/rpc/service/WEB-INF/classes/localhost
```

署名付与の設定情報について

- 署名付与に必要な情報は，Web サービスセキュリティ機能定義ファイルに指定された設定情報から取得します。
- 署名に用いる秘密鍵は，環境設定ファイルで指定したキーストアファイルを使用します。環境設定ファイルの詳細については，「3.8.2　環境設定ファイルの設定項目」を参照してください。
- 署名に関する仕様のサポート範囲については，「付録 A.2　XML 署名標準仕様のサポート範囲」を参照してください。

署名検証の設定情報について

- 署名検証に必要な情報は，Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイルに指定された設定情報から取得します。
- 証明書検証に用いる証明書は，環境設定ファイルで指定した証明書ファイルを使用します。環境設定ファイルの詳細については，「3.8.2　環境設定ファイルの設定項目」を参照してください。

注意事項

署名付与／検証で使用するキーストアファイルは，読み取り権限を限定するなど，第三者に読み取られないようにご注意ください。

3.2.1 署名を付与する個所をパート名で指定する

署名を付与する個所をパート名という名称で指定します。Cosminexus Web Services - Security で，署名を付与する際に指定できるパート名は，"Body" だけです。"Body" を指定した場合，SOAP メッセージの Body 要素そのものを意味します。

署名個所をパート名で指定する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
SecurityConfig
└┬ RequestSenderConfig/ResponseSenderConfig
 └┬ SenderPortConfig
  └┬ RoleConfig
   └┬ SignatureConfig
    └─ SignatureTarget
```

署名個所をパート名で指定する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
 └┬ ReceiverPortConfig
  └┬ VerificationConfig
   └─ SignatureTarget
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

3.2.2 署名を付与する個所を ID 属性で指定する

署名を付与する個所を，ID 属性で指定し，その ID 値が含まれる要素に対して署名します。なお，受信側で ID 属性のポリシーについては検証できません。

署名個所を ID で指定する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
SecurityConfig
└┬ RequestSenderConfig/ResponseSenderConfig
 └┬ SenderPortConfig
  └┬ RoleConfig
   └┬ SignatureConfig
    └─ SignatureTarget
```

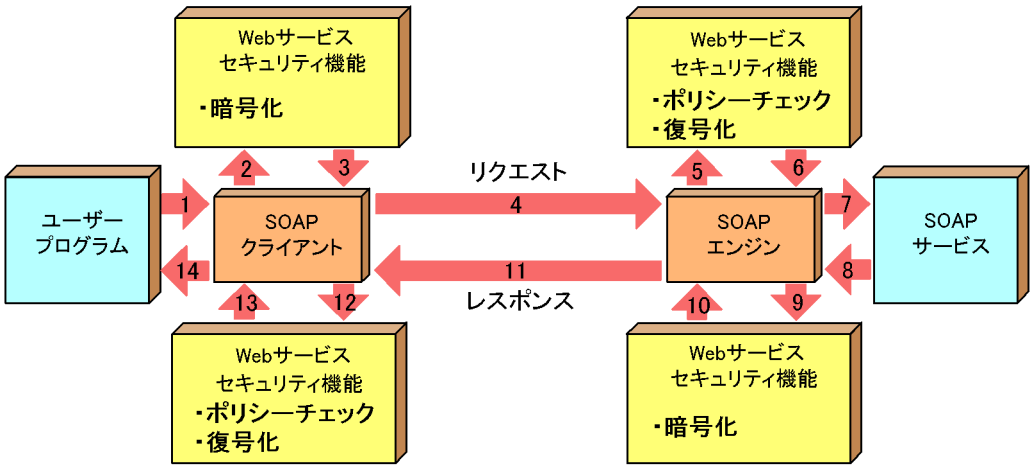
設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

3.3　暗号化／復号化機能を設定する

暗号化／復号化機能は，SOAP 通信基盤で送受信する SOAP メッセージを，暗号化／復号化する機能です。暗号化機能は SOAP メッセージの送信時に，復号化機能は SOAP メッセージの受信時に使用します。

次に，暗号化／復号化機能を使用した際のプログラムの構成図を示します。矢印およびその番号は，送受信するデータの処理の流れを示しています。

図 3-2　Web サービスセキュリティ機能使用時のプログラム構成（暗号化／復号化）



暗号化／復号化機能を使用する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
＜送信側＞
SecurityConfig
├── BindingConfig
└┬ RequestSenderConfig/ResponseSenderConfig
 └┬ SenderPortConfig
  └┬ RoleConfig
   └── EncryptionConfig
＜受信側＞
SecurityConfig
├── BindingConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
 └┬ ReceiverPortConfig
  └── DecryptionConfig
```

暗号化／復号化機能を使用する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
 └┬ ReceiverPortConfig
  └── DecryptionPolicyConfig
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

Windows の場合，定義ファイルのサンプルは，次のディレクトリに格納されていますので，これらを参考に定義ファイルを作成してください。なお，作成した定義ファイルの格納場所については，「3.9　Web サービスセキュリティ機能の実装手順」を参照してください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
encryption/message/client/WEB-INF/classes
encryption/message/service/WEB-INF/classes
encryption/rpc/client/WEB-INF/classes
encryption/rpc/service/WEB-INF/classes
```

また，Windows の場合，コーディングのサンプルは，次のディレクトリに格納されていますので，参考にしてください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
encryption/SecretKey
encryption/message/client/WEB-INF/classes/messagesampleclient
encryption/message/service/WEB-INF/classes/messagesampleservice
encryption/rpc/client/WEB-INF/classes/localhost
encryption/rpc/service/WEB-INF/classes/localhost
```

暗号化の設定情報について

- 暗号化に必要な情報は，Web サービスセキュリティ機能定義ファイルに指定された設定情報から取得します。
- 暗号化に関する仕様のサポート範囲については，「付録 A.3　XML 暗号標準仕様のサポート範囲」を参照してください。

復号化の設定情報について

復号化に必要な情報は，Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイル指定された設定情報から取得します。

暗号化／復号化に用いる共通鍵は，環境設定ファイルで指定した共通鍵ファイルを使用します。環境設定ファイルの詳細については，「3.8.2　環境設定ファイルの設定項目」を参照してください。

注意事項

暗号化／復号化で使用する鍵ファイルは，読み取り権限を限定するなど，第三者に読み取られないようにご注意ください。

3.3.1　暗号化する個所をパート名で指定する

暗号化する個所をパート名という名称で指定します。Cosminexus Web Services - Security で，暗号化する際に指定できるパート名は，"BodyContent" だけです。"BodyContent" を指定した場合，SOAP メッセージの Body 要素に含まれる子要素，テキストなどをすべて指定することを意味します。

暗号個所をパート名で指定する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
SecurityConfig
└┬ RequestSenderConfig/ResponseSenderConfig
 └┬ SenderPortConfig
  └┬ RoleConfig
   └┬ EncryptionConfig
    └┬ ContentsEncryption
     └─ EncryptionTarget
```

暗号個所をパート名で指定する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
 └┬ ReceiverPortConfig
  └┬ DecryptionPolicyConfig
   └─ DecryptionTarget
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

3.3.2　暗号化する個所を ID 属性で指定する

暗号化する個所を，ID 属性で指定し，その ID 値が含まれる要素に対して暗号化します。なお，受信側で ID 属性のポリシーについては検証できません。

暗号個所を ID で指定する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
SecurityConfig
└┬ RequestSenderConfig/ResponseSenderConfig
 └┬ SenderPortConfig
  └┬ RoleConfig
   └┬ EncryptionConfig
    └┬ ContentsEncryption
     └─ EncryptionTarget
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

3.4 認証機能を設定する

セキュリティトークンを通信先に受け渡すことで，認証を実現します。

Cosminexus Web Services - Security では，JAAS を使用した認証方式をサポートします。

認証機能を使用する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
＜送信側＞
SecurityConfig
├── BindingConfig
└┬ RequestSenderConfig/ResponseSenderConfig
  └┬ SenderPortConfig
    └┬ RoleConfig
      └── UsernameTokenConfig
＜受信側＞
SecurityConfig
├── BindingConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
  └┬ ReceiverPortConfig
    └── AuthenticationConfig
```

認証機能を使用する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
├── GlobalConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
  └┬ ReceiverPortConfig
    └┬ SecurityTokenConfig
      └── UsernameTokenConfig
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

Windows の場合，定義ファイルのサンプルは，次のディレクトリに格納されていますので，これらを参考に定義ファイルを作成してください。なお，作成した定義ファイルの格納場所については，「3.9　Web サービスセキュリティ機能の実装手順」を参照してください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
usernameToken/rpc/client/WEB-INF/classes
usernameToken/rpc/service/WEB-INF/classes
```

また，Windows の場合，コーディングのサンプルは，次のディレクトリに格納されていますので，参考にしてください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
usernameToken/rpc/Auth
usernameToken/rpc/client/WEB-INF/classes/localhost
usernameToken/rpc/service/WEB-INF/classes/localhost
```

認証の設定情報について

認証に必要な情報は，Web サービスセキュリティ機能定義ファイルに指定された設定情報から取得します。ただし，一部の情報は Web サービスセキュリティ機能が提供する API を使用して指定できます。API については，「5. Web サービスセキュリティ機能が提供する API」を参照してください。

認証の SOAP メッセージ形式について

Web サービスセキュリティ機能が生成する SOAP メッセージ内の UsernameToken 要素の Nonce 要素と Created 要素は，パスワードのタイプがダイジェスト形式の場合に付与されます。パスワードタイプがテキスト形式の場合はこれらの要素は付与されません。

パスワードのタイプがダイジェスト形式の SOAP メッセージでダイジェストを求めたい場合は，コーディングのサンプルのソースコード（UsernameLoginModule.java）の login メソッドの実装を参考にしてください。

認証に関するポリシーチェックについて

Web サービスセキュリティポリシー定義ファイルに指定された設定情報を基に，受信した SOAP メッセージがポリシーに従っているかどうかをチェックします。

注意事項

認証で，パスワードの記述されたファイルなどを使用する場合は，読み取り権限を限定するなど，第三者に読み取られないようにご注意ください。

3.5 メッセージに有効期限を設定する

Cosminexus Web Services - Security では，SOAP メッセージの送信時にタイムスタンプや有効期限を設定したり，受信側で古いメッセージや有効期限を経過したメッセージの受信を拒否したりする機能を提供します。

メッセージに有効期限を設定する際の，Web サービスセキュリティ機能定義ファイルの設定項目を次に示します。

```
SecurityConfig
└┬ RequestSenderConfig/ResponseSenderConfig
  └┬ SenderPortConfig
    └┬ RoleConfig
      └┬ TimestampConfig
        └── Expires
```

メッセージに付与されているタイムスタンプや有効期限を検証する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
├── GlobalConfig
└┬ RequestReceiverConfig/ResponseReceiverConfig
  └┬ ReceiverPortConfig
    └┬ TimestampConfig
      └┬ Created
       └ Expires
```

設定する要素の詳細については，「付録 C　定義ファイルの項目の詳細」を参照してください。

Windows の場合，定義ファイルのサンプルは，次のディレクトリに格納されていますので，これらを参考に定義ファイルを作成してください。なお，作成した定義ファイルの格納場所については，「3.9　Web サービスセキュリティ機能の実装手順」を参照してください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
timeStamp/rpc/client/WEB-INF/classes
timeStamp/rpc/service/WEB-INF/classes
```

また，Windows の場合，コーディングのサンプルは，次のディレクトリに格納されていますので，参考にしてください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
timeStamp/rpc/client/WEB-INF/classes/localhost
timeStamp/rpc/service/WEB-INF/classes/localhost
```

3.6 定義ファイルの構文をチェックする

Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイルを設定したあと，定義ファイルの記述内容が正しいかどうかをチェックするために，構文チェック用のコマンドを使用します。

構文チェック用のコマンドの使い方については，「4.2　定義ファイル構文チェックコマンド（CWSSConfCheck）」を参照してください。

注意事項

定義ファイル構文チェックコマンドは，Windows で使用してください。その他の OS では使用できません。

3.7　定義ファイルに関する注意事項

Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイルを設定する上での注意事項について説明します。

(1)　Timestamp 要素を使用する場合

- Web サービスセキュリティ機能定義ファイルの Created 要素，Expires 要素を共に省略した場合は，Timestamp 要素自体が作成されません。
- SOAP メッセージを送信するマシンと受信するマシンに設定されている時刻は，一致していないおそれがあります。Created 要素に設定されている値を基にそのメッセージを古いとみなしたり，Expires 要素に設定されている値を基にそのメッセージを有効期限切れとみなしたりする際，許容範囲を設定できます。
  マシン間の設定時刻の差に関する許容範囲を設定する際の，Web サービスセキュリティポリシー定義ファイルの設定項目を次に示します。

```
PolicyConfig
└┬ GlobalConfig
  ├── Max-Clock-Skew
  └── Fresh-Time-Limit
```

詳細については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

(2)　Web サービスセキュリティ機能定義ファイルの運用について

- Web サービスセキュリティ機能定義ファイルは，第三者によって改ざんされないようにアクセス権の設定をしてください。

(3)　Web サービスセキュリティポリシー定義ファイルの運用について

- Web サービスセキュリティポリシー定義ファイルは，第三者によって改ざんされないようにアクセス権の設定をしてください。
- Web サービスセキュリティポリシー定義ファイルを SOAP サービスの利用者に配布する場合，ファイルが改ざんされないよう対策してください。
- Web サービスセキュリティポリシー定義ファイルは，安全な方法で SOAP サービスの利用者に配布してください。
- SOAP サービスの利用者は，Web サービスセキュリティポリシー定義ファイルの内容に従って，Web サービスセキュリティ機能定義ファイルを作成する必要があります。

(4)　セキュリティ機能を組み合わせて使用する場合

署名付与／検証機能，暗号化／復号化機能，認証機能，およびメッセージに有効期限を設定する機能は，組み合わせて使用できます。複数の機能を組み合わせて使用する場合は，Web サービスセキュリティ機能定義ファイルの RoleConfig 要素内に，それぞれの設定項目を併記してください。

次に，設定の順番についての注意事項を示します。

- 送信側の Web サービスセキュリティ機能定義ファイルに設定した機能は，設定した順番で処理されます。各機能に対応するセキュリティ要素が SOAP メッセージのセキュリティヘッダに出現しますが，その順番は，Web サービスセキュリティ機能定義ファイルで設定した順番とは逆順になります。
- Web サービスセキュリティ機能が提供する API を使用して UsernameToken を付与した場合，対応するセキュリティ要素は SOAP メッセージのセキュリティヘッダのいちばん下に出現します。

なお，受信側の Web サービスセキュリティ機能定義ファイルと，Web サービスセキュリティポリシー定義ファイルに設定する順番は，特に意識する必要はありません。

3.8 実行環境に合わせて設定を変更する

Web サービスセキュリティ機能定義ファイルおよび Web サービスセキュリティポリシー定義ファイルに記述する内容で，実行環境に依存する部分は，環境設定ファイルで設定します。

環境設定ファイルは，次のディレクトリに格納されています。ファイル名は，「cwsscfg.properties」で固定です。

＜ Cosminexus のインストールディレクトリ＞ /wss/conf の下

次に，環境設定ファイルの記述規則と設定項目について説明します。

3.8.1 環境設定ファイルの記述規則

環境設定ファイルは，Java™2 Platform Standard Edition のプロパティ形式に従い，「キー名称＝値」の形式で記述します。次に，環境設定ファイルの記述規則について示します。

- １行の終わりは必ず改行されていなければなりません。
- 「#」で始まる行はコメントとみなされます。
- 値が存在しない行を設定した場合，空文字として処理されます。
- キー名称と＝の間，および＝と値の間にスペースを入れてはいけません。
- 値に 2 バイト文字を使用する場合は，Cosminexus で提供されている Java™2SDK の native2ascii コマンドで環境設定ファイルを変換してから使用しなければなりません。
- 設定できるキー名称以外のキー名称を設定した場合，その行は無視されます。
- キー名称の大文字と小文字は区別されます。
- 値には半角スペースも設定できます。

  例

  ```
  cwss.binding.KeyLocator.KeyStoreDir=d:/Program Files/HITACHI/
  Cosminexus/wss/KeyStore
  ```

3.8.2 環境設定ファイルの設定項目

環境設定ファイルで設定できるキーおよび値を，次の表に示します。

表 3-3 環境設定ファイルの設定項目

キー名称	指定値	説明	デフォルト値（値省略時または範囲外）
cwss.binding.KeyLocator.KeyStoreDir	任意のディレクトリ名	署名付与／検証で使用するキーストアファイルを格納するディレクトリ名を指定します。	＜ Cosminexus のインストールディレクトリ＞ /wss/KeyStore
cwss.binding.KeyLocator.CertificateDir	任意のディレクトリ名	証明書検証で使用する証明書ファイルを格納するディレクトリ名を指定します。	＜ Cosminexus のインストールディレクトリ＞ /wss/Cert
cwss.policy.usernameToken.maxNonceCount	1 ～ 100,000	受信済み Nonce 値を保存する最大個数を指定します。	100,000
cwss.policy.usernameToken.maxNonceAge	1,000 ～ 86,400,000（最大 24 時間）	受信済み Nonce 値の最大保存時間を現在時刻からの相対ミリ秒で指定します。	300,000（5 分）
cwss.binding.KeyLocator.SecretKeyDir	任意のディレクトリ名	暗号化／復号化で使用する共通鍵ファイルを格納するディレクトリ名を指定します。	＜ Cosminexus のインストールディレクトリ＞ /wss/SecretKey
cwss.engine.receive.unsecured.message	true または false	受信側で Web サービスセキュリティポリシー定義ファイルの設定をしている場合，受信メッセージに SOAP メッセージの Security 要素を含まなかったときの動作を指定します。※	false

注※

受信側で Web サービスセキュリティポリシー定義ファイルの設定をしている場合で，false を設定した場合，エラーメッセージ KDCGF0003-E が出力されます。true を設定した場合，Web サービスセキュリティ機能定義ファイルおよび Web

サービスセキュリティポリシー定義ファイルの設定内容に関係なく，エラーにはなりません。

3.9 Web サービスセキュリティ機能の実装手順

ここでは，実行環境に必要な定義ファイルやユーザー作成のプログラムを実装し，開発した SOAP アプリケーションで Web サービスセキュリティ機能を使用する手順について説明します。

次に，SOAP アプリケーション開発支援機能を利用した SOAP アプリケーションの開発の流れを示します。

（a）サービスの設計

（b）WSDL の作成

（c）サーバスケルトンの作成

（d）サーバ実装

（e）WAR ファイルの作成

（f）サービスの開始

（g）クライアントスタブの生成

（h）クライアントの実装

SOAP アプリケーション開発の詳細については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

また，07-60 以降のバージョンを使用して Web サービスセキュリティ機能を使用する場合は，移行が必要です。移行の詳細については，「付録 B 下位バージョンからの移行手順」を参照してください。

前に示した SOAP アプリケーション開発の流れ（以降，「開発ステップ」と呼びます）を基本とし，Web サービスセキュリティ機能を使用する上で必要なステップを説明します。

3.9.1 サーバ側の実装手順

Web サービスセキュリティ機能をサーバ側に実装する手順について説明します。

（1） デプロイ定義ファイルにセキュリティ情報を付加する

SOAP アプリケーションの形態が，RPC または EJB の場合

開発ステップ「（c）サーバスケルトンの作成」で，Web サービスセキュリティ機能に関する情報※を，サーバ側 WAR ファイルの WEB-INF の下のデプロイ定義ファイルに手動で追加します。

SOAP アプリケーションの形態が，既存 Java クラスを利用，またはメッセージングの場合

開発ステップ「(e) WAR ファイルの作成」で，Web サービスセキュリティ機能に関する情報※をサーバ側 WAR ファイルの WEB-INF の下のデプロイ定義ファイルに手動で追加します。

注※

「Web サービスセキュリティ機能に関する情報」を次に示します。

```
<handler name="WSSResponseSenderHandler"
type="java:com.cosminexus.wss.handlers.WSSResponseSenderHandler
"/>
<handler name="WSSRequestReceiverHandler"
type="java:com.cosminexus.wss.handlers.WSSRequestReceiverHandle
r"/>

<requestFlow>
 <handler type="WSSRequestReceiverHandler"/>
</requestFlow>
<responseFlow>
 <handler type="WSSResponseSenderHandler"/>
</responseFlow>
```

Windows の場合，サーバ側のデプロイ定義ファイル（server-config.xml）のサンプルは，サンプルディレクトリの下の，機能ごとのサービスに格納されていますので，参考にしてください。次に，格納先の一例を示します。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

usernameToken/rpc/service/WEB-INF

(2) ユーザー作成のプログラムを実装する

Web サービスセキュリティ機能が提供する API を利用する場合は，開発ステップ「(d) サーバ実装」で，ユーザー作成のプログラムを実装します。詳細については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

(3) WAR ファイルを作成する

開発ステップ「(e) WAR ファイルの作成」で，Web サービスセキュリティ機能が提供する Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイルをサーバ側 WAR ファイルの WEB-INF/classes の下に組み込みます。Web サービスセキュリティ機能定義ファイルと Web サービスセキュリティポリシー定義ファイルについては，構文チェック用のコマンドを使用して，構文に誤りがないことを事前に確認してください。構文チェック用のコマンドの使い方については，「4.2 定義ファイル構文チェックコマンド（CWSSConfCheck)」を参照してください。

WAR ファイルへの組み込み方法については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

サーバ側の実装に関するファイルの配置は，各アプリケーションごと（WAR ファイルご

と）に次のようになります。

```
WEB-INF
 ├server-config.xml       デプロイ定義ファイル
 └classes
   ├security-config.xml   Webサービスセキュリティ機能定義ファイル
   └policy-config.xml     Webサービスセキュリティポリシー定義ファイル
```

3.9.2 クライアント側が Web アプリケーションの場合の実装手順

Web サービスセキュリティ機能をクライアント側の Web アプリケーションに実装する手順について説明します。

（1） デプロイ定義ファイルを組み込む

開発ステップ「(g) クライアントスタブの生成」で，Web サービスセキュリティ機能が提供するクライアント用のデプロイ定義ファイル（client-config.xml）を，クライアント側 WAR ファイルの WEB-INF/classes の下に組み込みます。Windows の場合，Web サービスセキュリティ機能が提供するクライアント用のデプロイ定義ファイルは，サンプルディレクトリの下の，機能ごとのクライアントに格納されています。どのクライアント用のデプロイ定義ファイルも内容は同じです。次に，格納先の一例を示します。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下
　usernameToken/rpc/client/WEB-INF/classes

（2） ユーザー作成のプログラムを実装する

Web サービスセキュリティ機能が提供する API を利用する場合は，開発ステップ「(h) クライアントの実装」で，ユーザー作成のプログラムを実装します。詳細については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

（3） WAR ファイルを作成する

サンプルファイルを基に作成した Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイルをクライアント側 WAR ファイルの WEB-INF/classes の下に組み込みます。Web サービスセキュリティ機能定義ファイルと Web サービスセキュリティポリシー定義ファイルについては，構文チェック用のコマンドを使用して，構文に誤りがないことを事前に確認してください。構文チェック用のコマンドの使い方については，「4.2　定義ファイル構文チェックコマンド (CWSSConfCheck)」を参照してください。

WAR ファイルへの組み込み方法については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

クライアント側の実装に関するファイルの配置は，各アプリケーションごと（WAR ファイルごと）に次のようになります。

```
WEB-INF
 └classes
   ├client-config.xml     デプロイ定義ファイル
   ├security-config.xml   Webサービスセキュリティ機能定義ファイル
   └policy-config.xml     Webサービスセキュリティポリシー定義ファイル
```

3.9.3　クライアント側がコマンドライン Java アプリケーションの場合の実装手順

サンプルファイルを基に作成した Web サービスセキュリティ機能定義ファイル，Web サービスセキュリティポリシー定義ファイル，およびデプロイ定義ファイルを，クライアント実行環境の CLASSPATH に設定したディレクトリの下に格納します。なお，Web サービスセキュリティ機能定義ファイルと Web サービスセキュリティポリシー定義ファイルについては，構文チェック用のコマンドを使用して，構文に誤りがないことを事前に確認してください。構文チェック用のコマンドの使い方については，「4.2　定義ファイル構文チェックコマンド（CWSSConfCheck）」を参照してください。

3.9.4　JAAS ログインモジュールの実装時の注意

ここでは，JAAS ログインモジュールを実装する際の注意事項について説明します。なお，JAAS ログインモジュールはユーザー側で開発します。一般的な JAAS ログインモジュールの開発方法については，Sun Microsystems,Inc. が公開している JAAS 認証に関するリファレンスガイドを参照してください。

（1）　Callback オブジェクトの生成について

JAAS ログインモジュールで認証情報を取得するためには，Web サービスセキュリティ機能が提供するコールバックハンドラを使用します。次に，JAAS ログインモジュール内で必要な処理を示します。

LoginModule.initialize

コールバックハンドラ（javax.security.auth.callback.CallbackHandler の実装クラス）をクラスメンバ変数に保持します。

LoginModule.login

コールバックハンドラ内で使用する Callback オブジェクト（javax.security.auth.callback.Callback）の配列を生成して，コールバックハンドラの handle メソッドを呼び出します。handle メソッドを呼び出したあと，Callback オブジェクトから認証情報を取り出して認証処理を行います。Callback オブジェクトの配列は，次の表に示す構成で生成してください。

表 3-4 生成する Callback オブジェクトの配列

配列番号	Callback オブジェクト	用途
0	javax.security.auth.callback.NameCallback	ユーザー名の取得
1	javax.security.auth.callback.PasswordCallback	パスワードの取得
2	javax.security.auth.callback.TextInputCallback	パスワード形式の取得
3	javax.security.auth.callback.TextInputCallback	Nonce 値の取得
4	javax.security.auth.callback.TextInputCallback	Created 値の取得

配列要素の取得方法を次に示します。

0：ユーザー名の取得

ユーザー名は javax.security.auth.callback.NameCallback.getName() メソッドで取得します。

1：パスワードの取得

パスワードは javax.security.auth.callback.PasswordCallback.getPassword() メソッドで取得します。パスワード形式がテキスト形式の場合，パスワードテキスト文字配列を返します。パスワード形式がダイジェスト形式の場合，ダイジェスト形式の文字配列を返します。パスワード省略時は null を返します。

2：パスワード形式の取得

パスワード形式は javax.security.auth.callback.TextInputCallback.getText() メソッドで取得します。パスワードの形式は次の文字列で返します。

"PasswordText"：テキスト形式

"PasswordDigest"：ダイジェスト形式

3：Nonce 値の取得

Nonce 値は javax.security.auth.callback.TextInputCallback.getText() メソッドで取得します。

4：Created 値の取得

Created 値は javax.security.auth.callback.TextInputCallback.getText() メソッドで取得します。

Windows の場合，JAAS ログインモジュールのコーディング例については，次のディレクトリに格納されているサンプルを参照してください。

＜ Cosminexus のインストールディレクトリ＞ /wss/samples/ の下

```
usernameToken/rpc/Auth
usernameToken/rpc/client/WEB-INF/classes/localhost
usernameToken/rpc/service/WEB-INF/classes/localhost
```

(2) ログイン構成ファイルの配置について

Cosminexus Component Container のユーザー定義ファイル（usrconf.properties）に，次のエントリーを追加してください。

java.security.auth.login.config== ＜ログイン構成ファイルのフルパス名＞

ユーザー定義ファイルについては，マニュアル「Cosminexus　リファレンス　定義編」を参照してください。

4 Web サービスセキュリティ機能が提供するコマンド

この章では，Web サービスセキュリティ機能が提供するコマンドの使い方について説明します。

4.1 共通鍵生成コマンド（CWSSCreateSecretKey）

共通鍵生成コマンドを使用して，暗号化機能を利用するときに必要な共通鍵を生成します。共通鍵生成コマンドを実行すると，共通鍵生成コマンドのオプションで指定したファイルに共通鍵が出力されます。共通鍵のファイルはバイナリー形式です。共通鍵生成コマンドの所在は次のとおりです。

＜ Cosminexus のインストールディレクトリ＞ /wss/bin/ の下

出力された共通鍵の扱い方については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」の SecretKeyFile 要素を参照してください。

（1） 形式

共通鍵生成コマンドの形式を次に示します。

```
CWSSCreateSecretKey.bat -h | -a <アルゴリズム識別子> -o <出力ファイル名>
```

コマンド名（CWSSCreateSecretKey.bat）のあとに，一つ以上の空白を挿入し，オプションを指定します。オプションと指定する値の間も一つ以上の空白を挿入します。

（2） オプション

共通鍵生成コマンドで指定するオプションを次に示します。

表 4-1　共通鍵生成コマンドのオプション

オプション	説明
-h	コマンドのオプション説明を表示します。
-a	生成する共通鍵のアルゴリズム識別子を指定します。 指定可能な値を次に示します。 "TRIPLEDES"：Triple DES ブロック暗号の共通鍵を生成します。 "AES128"：AES-128 ブロック暗号の共通鍵を生成します。
-o	生成した共通鍵を出力するファイル名を指定します。

（3） メッセージ

コマンド実行時のメッセージについては，「7.2.5　KDCGK で始まるメッセージ」を参照してください。

（4） 注意事項

- OS が Windows（x86 および x64）の場合だけ実行できます。ほかの OS では実行できません。
- 出力ファイル名に空白が含まれる場合は出力ファイル名を「"」で囲みます。

- 出力ファイル名にディレクトリを含まない場合は，カレントディレクトリに出力されます。
- 出力ファイル名にディレクトリを含めた場合は，既存のディレクトリを指定する必要があります。
- 出力ファイル名には既存のファイルを指定してはいけません。
- 同じオプションを 2 回以上指定すると，最後に指定したオプションの値が有効になります。次の例では，共通鍵は file2 に出力されます。

例

```
CWSSCreateSecretKey.bat -a AES128 -o file1 -o file2
```

4.2 定義ファイル構文チェックコマンド（CWSSConfCheck）

Web サービスセキュリティ機能定義ファイル，および Web サービスセキュリティポリシー定義ファイルを作成したあと，設定した XML の構文が正しいかどうかをチェックするために，定義ファイル構文チェックコマンドを使用します。定義ファイル構文チェックコマンドの所在は次のとおりです。

＜ Cosminexus のインストールディレクトリ＞ /wss/bin/ の下

（1） 形式

定義ファイル構文チェックコマンドの形式を次に示します。

```
CWSSConfCheck.bat -h | [-s | -p] -f <チェックする定義ファイル名>
```

コマンド名（CWSSConfCheck.bat）のあとに，一つ以上の空白を挿入し，オプションを指定します。オプションと指定する値の間も一つ以上の空白を挿入します。

（2） オプション

構文チェックコマンドで指定するオプションを次に示します。

表 4-2 定義ファイル構文チェックコマンドのオプション

オプション	説明
-h	コマンドのオプション説明を表示します。
-s	-f オプションで指定するファイルが，Web サービスセキュリティ機能定義ファイルの場合に指定します。
-p	-f オプションで指定するファイルが，Web サービスセキュリティポリシー定義ファイルの場合に指定します。
-f	チェック対象の定義ファイル名を指定します。

（3） 終了コード

構文チェックコマンドの終了コードを次に示します。

- 0：正常終了
- 1：エラー終了

（4） メッセージ

コマンド実行時のメッセージについては，「7.2.6 KDCGO で始まるメッセージ」を参照してください。

(5) 注意事項

- OS が Windows（x86 および x64）の場合だけ実行できます。ほかの OS では実行できません。
- 入力ファイル名に空白が含まれる場合は入力ファイル名を「"」で囲みます。
- 構文チェックで不正な構文があった場合，または内部エラーが発生した場合，エラーメッセージが標準出力に表示されます。
- 同じオプションを 2 回以上指定すると，最後に指定したオプションの値が有効になります。次の例では，file2 に対してだけ構文をチェックします。

例

```
CWSSConfCheck -s -f file1.xml -f file2.xml
```

5 Web サービスセキュリティ機能が提供する API

この章では，Web サービスセキュリティ機能が提供する API について説明します。

5.1 インタフェースおよびクラスの一覧

Web サービスセキュリティ機能が提供するインタフェースおよびクラスの一覧を次に示します。

表 5-1 提供するインタフェースおよびクラスの一覧

インタフェース名およびクラス名	説明
WSSElementProxyFactory	セキュリティ項目操作クラスのファクトリクラス
WSSElementProxy	セキュリティ項目操作クラス
WSSUsernameToken	UsernameToken 要素の操作クラス
WSSUsernameToken.PasswordType	UsernameToken 要素のパスワード種別を示す列挙定数
WSSException	例外情報を保持するクラス

5.2 WSSElementProxyFactory クラス（セキュリティ項目操作クラスの生成）

セキュリティ項目操作クラスのファクトリクラスです。

クラス定義

```
public final class WSSElementProxyFactory
```

パッケージ名

```
com.cosminexus.wss.element
```

WSSElementProxyFactory クラスのメソッドを次の表に示します。

表 5-2 WSSElementProxyFactory クラスのメソッド一覧

メソッド	機能概要
newWSSElementProxy（スタブクラスから生成）	スタブクラスから空のセキュリティ項目操作クラスのインスタンスを生成します。
newWSSElementProxy（メッセージクラスから生成）	メッセージ送信クラスから空のセキュリティ項目操作クラスのインスタンスを生成します。
newWSSElementProxy（実装クラスから生成）	空のセキュリティ項目操作クラスのインスタンスを生成します。
getWSSElementProxy（スタブクラスから生成）	スタブクラスから，受信したメッセージのセキュリティ項目操作クラスのインスタンスを生成します。
getWSSElementProxy（メッセージクラスから生成）	メッセージ送信クラスから，受信したメッセージのセキュリティ項目操作クラスのインスタンスを生成します。
getWSSElementProxy（実装クラスから生成）	受信したメッセージのセキュリティ項目操作クラスのインスタンスを生成します。

次に，各メソッドの詳細について説明します。

newWSSElementProxy（スタブクラスから生成）

クラス名：WSSElementProxyFactory

機能

スタブクラスから空のセキュリティ項目操作クラスのインスタンスを生成します。このメソッドの引数で指定したクライアントのインタフェースクラスのサービスメソッドを呼び出すと，セキュリティ項目操作クラスに設定した内容でセキュリティ要素を生成します。このメソッドは，呼び出すサービスメソッドの形態が RPC または EJB の場合に使用します。

構文

```
public static WSSElementProxy newWSSElementProxy (
     javax.xml.rpc.Stub a_Stub
) throws WSSException;
```

引数

表 5-3　newWSSElementProxy（スタブクラスから生成）メソッドの引数

仮引数名	名称	in/out	説明
a_Stub	スタブクラス	in	クライアントのインタフェースクラス（スタブクラス）を指定します。

戻り値

セキュリティ項目操作クラスのインスタンスです。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

- メソッドの引数の指定とは異なるクライアントのインタフェースクラスのサービスメソッドを呼び出した場合，セキュリティ項目操作クラスに設定した内容は反映しません。Web サービスセキュリティ機能定義ファイルに設定した内容に従って，セキュリティ項目を設定します。
- メソッドで取得したセキュリティ項目操作クラスが提供するメソッドを利用しないでサービスメソッドを呼び出した場合，Web サービスセキュリティ機能定義ファイルに設定した内容に従って，セキュリティ項目を設定します。
- 送信時に Web サービスセキュリティ機能を利用しない場合は，このメソッドで取得したセキュリティ項目操作クラスが提供するメソッドを設定しても，セキュリティ項目は SOAP メッセージに設定しません。Web サービスセキュリティ機能を利用しない場合を次に示します。

- Web サービスセキュリティ機能定義ファイルが送信時にデプロイされていない場合
- Web サービスセキュリティ機能定義ファイル中のリクエストメッセージ送信時の設定が省略されている場合

newWSSElementProxy（メッセージクラスから生成）

クラス名：WSSElementProxyFactory

機能

メッセージ送信クラスから空のセキュリティ項目操作クラスのインスタンスを生成します。このメソッドの引数で指定した SOAPMessageSender クラスの sendMessage メソッドを呼び出すと，セキュリティ項目操作クラスに設定した内容でセキュリティ要素を生成します。このメソッドは，呼び出すサービスメソッドの形態がメッセージングの場合に使用します。

構文

```
public static WSSElementProxy newWSSElementProxy (
     com.cosminexus.c4web.service.message.SOAPMessageSender
a_Sender
) throws WSSException;
```

引数

表 5-4　newWSSElementProxy（メッセージクラスから生成）メソッドの引数

仮引数名	名称	in/out	説明
a_Sender	メッセージ送信クラス	in	SOAP 通信基盤が提供する SOAPMessageSender クラスを指定します。

戻り値

セキュリティ項目操作クラスのインスタンスです。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

- SOAP 通信基盤の標準モードではこのメソッドは使用できません。呼び出すサービスメソッドの形態がメッセージングのときにユーザプログラムからセキュリティヘッダにアクセスしたい場合は，SOAP 通信基盤の互換モードでこのメソッドを使用します。SOAP 通信基盤のモードについては，マニュアル「Cosminexus　SOAP　アプリケーション開発ガイド」を参照してください。
- メソッドの引数の指定とは異なるインスタンスの SOAPMessageSender クラスの sendMessage メソッドを呼び出した場合，セキュリティ項目操作クラスに設定した内容は反映しません。Web サービスセキュリティ機能定義ファイルに設定した内容に従って，セキュリティ項目を設定します。
- メソッドで取得したセキュリティ項目操作クラスが提供するメソッドを利用しないで

sendMessage メソッドを呼び出した場合，Web サービスセキュリティ機能定義ファイルに設定した内容に従って，セキュリティ項目を設定します。

- 送信時に Web サービスセキュリティ機能を利用しない場合は，このメソッドで取得したセキュリティ項目操作クラスが提供するメソッドを利用しても，セキュリティ項目は SOAP メッセージに設定しません。Web サービスセキュリティ機能を利用しない場合を次に示します。
  - Web サービスセキュリティ機能定義ファイルが送信時にデプロイされていない場合
  - Web サービスセキュリティ機能定義ファイル中のリクエストメッセージ送信時の設定が省略されている場合

newWSSElementProxy（実装クラスから生成）

クラス名：WSSElementProxyFactory

機能

空のセキュリティ項目操作クラスのインスタンスを生成します。このメソッドは，スタブまたはメッセージクラスから生成するメソッドとは異なり，SOAP サービスの実装クラスでサービスの応答を送信する際に使用します。このメソッドで取得したセキュリティ項目操作クラスが提供するメソッドを利用すると，サービスの応答をクライアントに送信する際に，セキュリティ項目操作クラスに設定した内容でセキュリティ要素を生成します。

構文

```
public static WSSElementProxy newWSSElementProxy (
) throws WSSException;
```

引数

ありません。

戻り値

セキュリティ項目操作クラスのインスタンスです。

例外

WSSException
　処理中に予測しない例外が発生した場合にスローされます。

注意事項

- メソッドで取得したセキュリティ項目操作クラスが提供するメソッドを利用しない場合，Web サービスセキュリティ機能定義ファイルに設定した内容に従って，レスポンスメッセージ内のセキュリティ項目を設定します。
- メソッドをリクエスト受信処理以外の場所で呼び出した場合は null を返します。
- 送信時に Web サービスセキュリティ機能を利用しない場合は，このメソッドで取得したセキュリティ項目操作クラスが提供するメソッドを利用しても，セキュリティ項目は設定しません。Web サービスセキュリティ機能を利用しない場合を次に示します。
  - Web サービスセキュリティ機能定義ファイルが送信時にデプロイされていない場合
  - Web サービスセキュリティ機能定義ファイル中のリクエストメッセージ送信時の設定が省略されていた場合

getWSSElementProxy（スタブクラスから生成）

クラス名：WSSElementProxyFactory

機能

スタブクラスから，受信したメッセージのセキュリティ項目操作クラスのインスタンスを生成します。このメソッドの引数で指定したクライアントのインタフェースクラスのサービスメソッドの呼び出し後にこのメソッドを呼び出すと，サービスメソッドのレスポンスメッセージに含まれるセキュリティ項目を取得し，セキュリティ項目操作クラスを生成します。その後，このメソッドで取得したセキュリティ項目操作クラスのメソッドを呼び出すことで，セキュリティ項目を取得できます。

構文

```
public static WSSElementProxy[] getWSSElementProxy (
     javax.xml.rpc.Stub a_Stub
) throws WSSException;
```

引数

表 5-5　getWSSElementProxy（スタブクラスから生成）メソッドの引数

仮引数名	名称	in/out	説明
a_Stub	スタブクラス	in	RPC または EJB を利用した SOAP アプリケーションの場合で，SOAP アプリケーション開発支援機能によって作成されるクライアントのインタフェースクラス（スタブクラス）を指定します。

戻り値

セキュリティ項目操作クラスのインスタンス配列です。セキュリティ項目操作クラスが生成できない場合は null を返します。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

- このメソッドは，引数に指定したスタブクラスのサービスメソッドの呼び出し直後に必ず呼び出すようにしてください。引数に指定したものと異なるスタブクラスのサービスメソッド呼び出し直後にこのメソッドを呼び出した場合，引数に指定したスタブクラスのサービスメソッドの応答メッセージに含まれるセキュリティ項目は取得されません。
- メソッドで取得するセキュリティ項目は，このメソッドを呼び出す直前に呼び出したサービスメソッドのレスポンスメッセージに含まれるセキュリティ項目です。サービ

スメソッドを複数回呼び出した場合，最後のサービスメソッド呼び出しのレスポンスメッセージに含まれるセキュリティ項目を取得します。サービスメソッドを一度も呼び出さないで，このメソッドを呼び出した場合は null を返します。

- 戻り値であるセキュリティ項目クラスの配列の順序と，レスポンスメッセージに含まれるセキュリティ項目の順序は必ずしも一致しません。

getWSSElementProxy（メッセージクラスから生成）

クラス名：WSSElementProxyFactory

機能

メッセージ送信クラスから，受信したメッセージのセキュリティ項目操作クラスのインスタンスを生成します。このメソッドの引数で指定した SOAPMessageSender クラスの sendMessage メソッドの呼び出し後にこのメソッドを呼び出すと，sendMessage メソッドのレスポンスメッセージに含まれるセキュリティ項目を取得し，セキュリティ項目操作クラスを生成します。その後，このメソッドで取得したセキュリティ項目操作クラスのメソッドを呼び出すことで，セキュリティ項目を取得できます。

構文

```
public static WSSElementProxy[] getWSSElementProxy (
     com.cosminexus.c4web.service.message.SOAPMessageSender
a_Sender
) throws WSSException;
```

引数

表 5-6　getWSSElementProxy（メッセージクラスから生成）メソッドの引数

仮引数名	名称	in/out	説明
a_Sender	メッセージ送信クラス	in	メッセージングを利用した SOAP アプリケーションの場合で，SOAP アプリケーション開発支援機能が提供する SOAPMessageSender クラスを指定します。

戻り値

セキュリティ項目操作クラスのインスタンス配列です。セキュリティ項目操作クラスが生成できない場合は null を返します。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

- SOAP 通信基盤の標準モードではこのメソッドは使用できません。呼び出すサービスメソッドの形態がメッセージングのときにユーザプログラムからセキュリティヘッダにアクセスしたい場合は，SOAP 通信基盤の互換モードでこのメソッドを使用します。SOAP 通信基盤のモードについては，マニュアル「Cosminexus　SOAP　アプリケーション開発ガイド」を参照してください。
- メソッドで取得するセキュリティ項目は，このメソッドを呼び出す直前に呼び出したサービスメソッドのレスポンスメッセージに含まれるセキュリティ項目です。サービ

スメソッドを複数回呼び出した場合，最後のサービスメソッド呼び出しのレスポンスメッセージに含まれるセキュリティ項目を取得します。サービスメソッドを一度も呼び出さないで，このメソッドを呼び出した場合は null を返します。

- 戻り値であるセキュリティ項目クラスの配列の順序と，レスポンスメッセージに含まれるセキュリティ項目の順序は必ずしも一致しません。

getWSSElementProxy（実装クラスから生成）

クラス名：WSSElementProxyFactory

機能

受信したメッセージのセキュリティ項目操作クラスのインスタンスを生成します。このメソッドは，スタブまたはメッセージクラスから生成するメソッドとは異なり，SOAP サービスの実装クラスでサービスのリクエストメッセージに含まれるセキュリティ項目を取得する際に使用します。SOAP サービスの実装クラスでこのメソッドを呼び出すと，リクエストメッセージに含まれるセキュリティ項目を取得し，セキュリティ項目操作クラスを生成します。その後，このメソッドで取得したセキュリティ項目操作クラスのメソッドを呼び出すことで，セキュリティ項目を取得できます。

構文

```
public static WSSElementProxy[] getWSSElementProxy (
) throws WSSException;
```

引数

ありません。

戻り値

セキュリティ項目操作クラスのインスタンス配列です。セキュリティ項目操作クラスが生成できない場合は null を返します。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

- このメソッドで取得するセキュリティ項目は，Web サービスセキュリティ機能が処理するセキュリティ項目です。
- 戻り値であるセキュリティ項目クラスの配列の順序と，リクエストメッセージに含まれるセキュリティ項目の順序は必ずしも一致しません。

5.3 WSSElementProxy クラス（セキュリティ項目の操作）

セキュリティ項目の操作クラスです。

クラス定義

```
public final class WSSElementProxy
```

パッケージ名

```
com.cosminexus.wss.element
```

WSSElementProxy クラスのメソッドを次の表に示します。

表 5-7 WSSElementProxy クラスのメソッド一覧

メソッド名称	説明
getWSSUsernameToken	UsernameToken 要素クラスのインスタンスを取得します。
setWSSUsernameToken	UsernameToken 要素クラスのインスタンスをセキュリティ項目操作クラスのインスタンスに設定します。
removeWSSUsernameToken	UsernameToken 要素クラスのインスタンスをセキュリティ項目操作クラスのインスタンスから削除します。
getRole	セキュリティ項目の role 属性を取得します。
setRole	セキュリティ項目の role 属性を設定します。

getWSSUsernameToken

クラス名：WSSElementProxy

機能

UsernameToken 要素クラスのインスタンスを取得します。

構文

```
public WSSUsernameToken[] getWSSUsernameToken (
) throws WSSException;
```

引数

ありません。

戻り値

UsernameToken 要素クラスのインスタンス配列です。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

- セキュリティ項目操作クラスのファクトリクラスの，newWSSElementProxy メソッドによって取得した WSSElementProxy クラスのインスタンスの場合，このメソッドの戻り値は null です。ただし，setWSSUsernameToken を呼んだ後，このメソッドを呼び出すと setWSSUsernameToken で指定した値を返します。
- 戻り値の UsernameToken 要素クラスのインスタンス配列の順序と，メッセージ中の UsernameToken 要素の順序とは必ずしも一致しません。

setWSSUsernameToken

クラス名：WSSElementProxy

機能

UsernameToken 要素クラスのインスタンスをセキュリティ項目操作クラスに設定します。

構文

```
public void setWSSUsernameToken (
     WSSUsernameToken a_UsernameToken
) throws WSSException;
```

引数

表 5-8　setWSSUsernameToken メソッドの引数

仮引数名	名称	in/out	説明
a_UsernameToken	UsernameToken 要素クラス	in	セキュリティ項目操作クラスに設定する UsernameToken 要素操作クラスのインスタンスを指定します。

戻り値

ありません。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

引数に null を指定した場合，セキュリティ項目操作クラスに UsernameToken 要素クラスを設定しないで正常終了します。すでに UsernameToken 要素クラスを設定している場合は，元の UsernameToken 要素クラスも更新しません。

removeWSSUsernameToken

クラス名：WSSElementProxy

機能

UsernameToken 要素クラスのインスタンスをセキュリティ項目操作クラスから削除します。

構文

```
public void removeWSSUsernameToken (
) throws WSSException;
```

引数

ありません。

戻り値

ありません。

例外

WSSException

処理中に予測しない例外が発生した場合にスローされます。

注意事項

UsernameToken 要素クラスのインスタンスがセキュリティ項目操作クラスに存在しない場合，何も処理をしないで正常終了します。

getRole

クラス名：WSSElementProxy

機能

セキュリティ項目の role 属性値を取得します。

構文

```
public java.net.URI getRole (
) throws WSSException;
```

引数

ありません。

戻り値

セキュリティ項目に含まれる role 属性値です。セキュリティ項目に role 属性がない場合は null を返します。

例外

ありません。

setRole

クラス名：WSSElementProxy

機能

セキュリティ項目の role 属性値を設定します。

構文

```
public void setRole (
     java.net.URI a_Role
) throws WSSException;
```

引数

表 5-9　setRole メソッドの引数

仮引数名	名称	in/out	説明
a_Role	role 属性値	in	セキュリティ項目操作クラスに設定する role 属性値を指定します。

戻り値

ありません。

例外

ありません。

5.4 WSSUsernameToken クラス（UsernameToken 要素の操作）

UsernameToken 要素の操作クラスです。

クラス定義

```
public final class WSSUsernameToken extends
     com.cosminexus.wss.element.WSSElementBase
```

パッケージ名

```
com.cosminexus.wss.element
```

WSSUsernameToken クラスのメソッドを次の表に示します。

表 5-10 WSSUsernameToken クラスのメソッド一覧

メソッド名称	説明
コンストラクタ	UsernameToken 要素クラスのコンストラクタです。
getUsername	ユーザー名を取得します。
setUsername	ユーザー名を設定します。
getPassword	パスワードを取得します。
setPassword	パスワードを設定します。
getId	UsernameToken 要素の Id 属性を取得します。
setId	UsernameToken 要素の Id 属性を設定します。
getPasswordType	パスワード種別を取得します。
setPasswordType	パスワード種別を設定します。
getNonce	Nonce 属性を取得します。
getCreated	Created 属性を取得します。

注意事項

- セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの newWSSElementProxy メソッドによって生成した場合，このクラスのメソッドで指定した内容で UsernameToken 要素を生成します。ただし，Web サービスセキュリティ機能定義ファイルに，呼び出すサービスメソッドに対応する SenderPortConfig 要素，RoleConfig 要素の指定がない場合は UsernameToken 要素を生成しません。
- セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの getWSSElementProxy メソッドによって生成した場合，このクラスの名称が set で始まるメソッドで値を設定しても，UsernameToken 要素は生成しません。
- Web サービスセキュリティ機能定義ファイルに UsernameToken 要素を定義している

場合，このメソッドが生成した UsernameToken 要素とは別に，複数の UsernameToken 要素を生成します。

コンストラクタ

機能

UsernameToken 要素クラスのコンストラクタです。

構文

```
public WSSUsernameToken (
     java.lang.String a_Username
) throws WSSException;
```

引数

表 5-11　コンストラクタの引数

仮引数名	名称	in/out	説明
a_Username	ユーザー名	in	ユーザー名を示す文字列を指定します。

戻り値

UsernameToken 要素クラスのインスタンスです。

例外

WSSException

引数に空文字または null が指定されました。

注意事項

このクラスが生成される際の UsernameToken 要素の初期値を次に示します。

表 5-12　UsernameToken 要素の初期値

クラスの要素	初期値 1 ※1	初期値 2 ※2
Username	コンストラクタの引数で指定したユーザー名	UsernameToken 要素のユーザー名
Password	null	UsernameToken 要素のパスワード
PasswordType	WSSUsernameToken.PasswordType.TEXT	WSSUsernameToken.PasswordType.TEXT または WSSUsernameToken.PasswordType.DIGEST
Id	null	UsernameToken 要素の Id 属性
Nonce	null	UsernameToken 要素の Nonce 属性
Created	null	UsernameToken 要素の Created 属性

注※ 1

コンストラクタを使用した場合

注※ 2

このクラスをセキュリティ項目操作クラスの getWSSUsernameToken メソッドで生成した場合

- 引数に null を設定した場合，WSSException 例外が発生します。
- 引数に空文字 "" を設定した場合，WSSException 例外が発生します。
- 引数に一つ以上の空白文字を設定した場合，ユーザー名の情報は一つ以上の空白文字に置き換えられます。

getUsername

クラス名：WSSUsernameToken

機能

UsernameToken 要素のユーザー名を取得します。

構文

```
public java.lang.String getUsername (
);
```

引数

ありません。

戻り値

UsernameToken 要素のユーザー名です。

例外

ありません。

setUsername

クラス名：WSSUsernameToken

機能

UsernameToken 要素のユーザー名を設定します。

構文

```
public void setUsername (
     java.lang.String a_Username
) throws WSSException;
```

引数

表 5-13　setUsername メソッドの引数

仮引数名	名称	in/out	説明
a_Username	ユーザー名	in	ユーザー名を示す文字列を指定します。

戻り値

ありません。

例外

WSSException

引数に空文字 "" が指定されました。

注意事項

- 引数に null を設定した場合，このクラスが保持するユーザー名の情報は更新しません。
- 引数に一つ以上の空白文字を設定した場合，このクラスが保持するユーザー名の情報は一つ以上の空白文字に置き換えられます。

getPassword

クラス名：WSSUsernameToken

機能

UsernameToken 要素のパスワードを取得します。

構文

```
public char[] getPassword (
);
```

引数

ありません。

戻り値

UsernameToken 要素のパスワードです。UsernameToken 要素にパスワードがない場合は null を返します。

例外

ありません。

注意事項

- セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの newWSSElementProxy メソッドによって生成した際，UsernameToken 要素内に Password 要素が存在しない場合は null を返します。ただし，すでに setPassword メソッドでパスワードを設定している場合はその値を返します。
- 受信メッセージのセキュリティ項目の，UsernameToken 要素内の Password 要素が空要素の場合は null を返します。
- セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの getWSSElementProxy メソッドによって生成した際，受信メッセージ内の Password 要素の種別が平文（SOAP メッセージの PasswordText 要素）の場合は平文のパスワードを返します。Password 要素の種別がダイジェスト（SOAP メッセージの PasswordDigest 要素）の場合はダイジェスト値をそのまま返します。

setPassword

クラス名：WSSUsernameToken

機能

UsernameToken 要素のパスワードを設定します。

構文

```
public void setPassword (
     char[] a_Password
);
```

引数

表 5-14　setPassword メソッドの引数

仮引数名	名称	in/out	説明
a_Password	パスワード	in	パスワードを指定します。

戻り値

ありません。

例外

ありません。

注意事項

- 引数に null または空文字 "" を設定した場合，このクラスに設定したパスワードの情報は更新されません。
- 引数に空白文字を設定した場合，このクラスに設定したパスワードの情報は空白文字に置き換えられます。

getId

クラス名：WSSUsernameToken

機能

UsernameToken 要素の Id 属性を取得します。

構文

```
public java.lang.String getId (
);
```

引数

ありません。

戻り値

UsernameToken 要素の Id 属性値を示す文字列です。UsernameToken 要素に Id 属性がない場合は null を返します。

例外

ありません。

注意事項

- このメソッドで取得するのは，UsernameToken 要素の Id 属性です。
- セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの newWSSElementProxy メソッドによって生成した場合は null を返します。ただし，すでに setId メソッドで Id を設定している場合はその値を返します。

setId

クラス名：WSSUsernameToken

機能

UsernameToken 要素の Id 属性を設定します。

構文

```
public void setId (
     java.lang.String a_Id
);
```

引数

表 5-15　setId メソッドの引数

仮引数名	名称	in/out	説明
a_Id	Id	in	Id 属性を示す文字列を指定します。

戻り値

ありません。

例外

ありません。

注意事項

このメソッドで設定した Id 属性値は UsernameToken 要素の Id 属性に設定されます。ただし，ほかの要素の Id 属性値と重複しているかどうかはチェックしません。

getPasswordType

クラス名：WSSUsernameToken

機能

UsernameToken 要素のパスワード種別を取得します。

構文

```
public WSSUsernameToken.PasswordType getPasswordType (
);
```

引数

ありません。

戻り値

UsernameToken 要素のパスワード種別です。UsernameToken 要素にパスワードがない場合は，WSSUsernameToken.PasswordType.TEXT を返します。

表 5-16　WSSUsernameToken.PasswordType クラスの列挙値

列挙値	意味
WSSUsernameToken.PasswordType.TEXT	平文形式のパスワードです。（http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-username-token-profile-1.0#PasswordText 形式）
WSSUsernameToken.PasswordType.DIGEST	ダイジェスト形式のパスワードです。（http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-username-token-profile-1.0#PasswordDigest 形式）

例外

ありません。

注意事項

セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの newWSSElementProxy メソッドによって生成した場合，WSSUsernameToken.PasswordType.TEXT を返します。ただし，すでに setPasswordType メソッドでパスワード種別を設定している場合は，その値を返します。

setPasswordType

クラス名：WSSUsernameToken

機能

UsernameToken 要素のパスワード種別を設定します。

構文

```
public void setPasswordType (
     WSSUsernameToken.PasswordType a_PasswordType
);
```

引数

表 5-17　setPasswordType メソッドの引数

仮引数名	名称	in/out	説明
a_PasswordType	パスワード種別	in	パスワード種別を示す，WSSUsernameToken.PasswordType の列挙値を指定します。

戻り値

ありません。

例外

ありません。

getNonce

クラス名：WSSUsernameToken

機能

UsernameToken 要素の Nonce 属性の内容を取得します。

構文

```
public java.lang.String getNonce (
);
```

引数

ありません。

戻り値

UsernameToken 要素の Nonce 属性値です。UsernameToken 要素に Nonce 属性がない場合は null を返します。

例外

ありません。

注意事項

セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの newWSSElementProxy メソッドによって生成した際，UsernameToken 要素内に Nonce 属性がない場合は null を返します。

getCreated

クラス名：WSSUsernameToken

機能

UsernameToken 要素の Created 属性の内容を取得します。

構文

```
public java.lang.String getCreated (
);
```

引数

ありません。

戻り値

UsernameToken 要素の Created 属性文字列です。UsernameToken 要素に Created 属性がない場合は null を返します。

例外

ありません。

注意事項

セキュリティ項目操作クラスのインスタンスを，セキュリティ項目操作クラスのファクトリクラスの newWSSElementProxy メソッドによって生成した際，UsernameToken 要素内に Created 属性がない場合は null を返します。

5.5 WSSUsernameToken.PasswordType インタフェース（PasswordType 要素の操作）

PasswordType 要素の操作クラスです。

クラス定義

```
public static final class WSSUsernameToken.PasssswordType
```

パッケージ名

```
com.cosminexus.wss.element
```

機能

WSSUsernameToken クラスの getPasswordType()，setPasswordType() メソッドの戻り値および引数で指定する，PasswordType 要素を示す列挙定数です。次にパスワード種別を示します。

TEXT

パスワード種別が平文形式（http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0#PasswordText）であることを示します。

DIGEST

パスワード種別がダイジェスト形式（http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-username-token-profile-1.0#PasswordDigest）であることを示します。

5.6　WSSException クラス（例外情報の取得）

Web サービスセキュリティ機能が提供する API で発生する例外クラスです。

クラス定義

```
public final class WSSException extends
     java.lang.Exception
```

パッケージ名

```
com.cosminexus.wss.faults
```

WSSException クラスのメソッドを次の表に示します。

表 5-18　WSSException クラスのメソッド一覧

メソッド名称	説明
getMessage	詳細メッセージを取得します。

getMessage

クラス名：WSSException

機能

例外の詳細メッセージを取得します。

構文

```
public java.lang.String getMessage (
);
```

引数

ありません。

戻り値

詳細メッセージを示す文字列です。

6 障害対策

この章では，Web サービスセキュリティ機能の実行，および運用中の障害対策のために出力される情報について説明します。

6.1 トレースを収集する

6.1 トレースを収集する

Web サービスセキュリティ機能では，障害対策に必要な情報をトレースとして出力します。トレースは障害の発生個所の割り出しや原因の調査などに利用します。Web サービスセキュリティ機能で出力するトレースの種類を次に示します。

表 6-1 Web サービスセキュリティ機能が出力するトレースの種類

トレースの種類	説明
サーバトレース	Web サービスセキュリティ機能のサーバ処理部分が出力します。
クライアントトレース	Web サービスセキュリティ機能のクライアント処理部分が出力します。
コマンドトレース	Web サービスセキュリティ機能のコマンドラインインタフェースが出力します。

6.1.1 トレースの内容

サーバトレース，クライアントトレース，およびコマンドトレースが出力するトレースの内容と出力例を次に示します。

(1) 出力内容

表 6-2 トレースの出力内容

項目	内容
日付	出力時の日付（yyyy/mm/dd 形式）が出力されます。
時刻	出力時の時刻（hh:mm:ss.sss 形式）が出力されます。
アプリケーション名	「wss」が出力されます。
プロセス識別子	プロセス識別子（16 進数）が出力されます。
スレッド識別子	スレッド識別子（16 進数）が出力されます。
メッセージ ID	メッセージ ID が出力されます。メッセージ ID を持たないものは出力されません。
メッセージ種別	OC：メソッドの入口を表します。 OD：メソッドの出口を表します。 EC：例外をキャッチしたことを表します。 ER：エラーメッセージを表示したことを表します。 FB：ほかのプログラムの処理の呼び出しを表します。 FE：呼び出したほかのプログラムの処理の終了を表します。 PB：Web サービスセキュリティ機能が提供する API の開始を表します。 PE：Web サービスセキュリティ機能が提供する API の終了を表します。
メッセージテキスト	実行時の情報を示すメッセージが出力されます。

(2) 出力例

図 6-1 トレースの出力例

```
yyyy/mm/dd hh:mm:ss.sss       pid  tid       message-id  message(LANG=0x0411)
0001 2002/01/17 18:09:54.224  wss  000006CC  J00737FE    0C  enter
WSSRequestSenderHandler::init()
0002 2002/01/17 18:09:54.224  wss  000006CC  J00737FE    0D  exit
WSSRequestSenderHandler::init()
```

6.1.2 トレースの出力先

トレースの出力先を次に示します。

表 6-3 トレースの出力先

トレースの種類	出力先
サーバトレース	SOAP 通信基盤のサーバ側で出力するトレース※と同じファイル上に出力されます。
クライアントトレース	SOAP 通信基盤のクライアント側で出力するトレース※と同じファイル上に出力されます。
コマンドトレース	SOAP 通信基盤のモード※により出力先ディレクトリが異なります。 • 標準モードの場合 ＜ Cosminexus のインストールディレクトリ＞ /CC/client/logs/system/ejbcl/WS • 互換モードの場合 ＜ Cosminexus のインストールディレクトリ＞ /wss/logs このディレクトリの下に次のファイル名で出力されます。 • 共通鍵生成コマンドの場合 CWSSCreateSecretKey-n.log (n:1 ～ 2) • 定義ファイル構文チェックコマンドの場合 CWSSConfCheck-n.log (n:1 ～ 2)

注※

SOAP 通信基盤のトレース出力先とモードについては，マニュアル「Cosminexus SOAP アプリケーション開発ガイド」を参照してください。

6.1.3 トレースの重要度

トレースの重要度を変更することで，出力するトレースの情報量を変更できます。出力する情報量を多くすることで，障害発生の要因を特定しやすくなります。ただし，出力する情報量を多くすると，プログラムの処理性能への影響が大きくなります。

トレースの重要度には次のレベルがあります。

- ERROR

- WARN
- INFO
- DEBUG

Web サービスセキュリティ機能で出力するトレースの重要度は，SOAP 通信基盤の設定に従います。詳細については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

7 メッセージ一覧

この章では，SOAP メッセージの送受信やコマンドの実行中などにエラーが発生した場合に出力されるメッセージの形式，および内容について説明します。定義ファイルの読み込み時，コマンド実行時，または API 使用時に出力されるメッセージについても説明します。なお，メッセージの説明では，メッセージはメッセージ ID 順に並んでいます。

7.1 メッセージの形式

メッセージはそれぞれ，プレフィックス，メッセージ番号，およびメッセージ種別から構成されるメッセージ ID を持っています。例えば，KDCGW0001-E というメッセージ ID の場合，「KDCGW」の部分がプレフィックス，「0001」の部分がメッセージ番号，「-E」の部分がメッセージ種別を表しています。この節では，メッセージのプレフィックスと種別について説明します。

(1) プレフィックス

メッセージのプレフィックスは，そのメッセージが出力されるタイミングによって異なります。Web サービスセキュリティ機能を利用する場合に出力されるメッセージのプレフィックス，各プレフィックスのメッセージが出力されるタイミング，および出力先を次の表に示します。

表 7-1 メッセージのプレフィックス一覧

項番	プレフィックス	出力のタイミング	出力先
1	KDCGA	Web サービスセキュリティ機能が提供する API で，エラーが発生した場合に出力されます。	KDCGF および KDCGW で始まるメッセージの詳細部分に出力されます。
2	KDCGC	Web サービスセキュリティ機能定義ファイルおよび Web サービスセキュリティポリシー定義ファイルを読み込む場合に，両定義ファイルに共通のエラーが発生したときに出力されます。	KDCGF，KDCGO，または KDCGW で始まるメッセージの詳細部分に出力されます。定義ファイル構文チェックコマンドの実行中は，標準出力および Web サービスセキュリティ機能のトレースファイルに出力されます。
3	KDCGF	SOAP メッセージを受信中にエラーが発生した場合に出力されます。	SOAP 通信基盤が提供する C4Fault クラスの faultString 要素として，ユーザー作成プログラムのメッセージに出力されます。メッセージの本文は，SOAP 通信基盤のトレースファイルにも出力されます。
4	KDCGJ	SOAP メッセージの認証でエラーが発生した場合に出力されます。	SOAP 通信基盤のトレースファイルに出力されます。
5	KDCGK	共通鍵生成コマンドの処理が正常に終了した場合，または実行中にエラーが発生した場合に出力されます。	標準出力および Web サービスセキュリティ機能のトレースファイルに出力されます。
6	KDCGO	定義ファイル構文チェックコマンドの処理が正常に終了した場合，または実行中にエラーが発生した場合に出力されます。	

項番	プレフィックス	出力のタイミング	出力先
7	KDCGP	Web サービスセキュリティポリシー定義ファイルの読み込み中にエラーが発生した場合に出力されます。	KDCGF，KDCGO，または KDCGW で始まるメッセージの詳細部分に出力されます。定義ファイル構文チェックコマンドの実行中は，標準出力および Web サービスセキュリティ機能のトレースファイルに出力されます。
8	KDCGS	Web サービスセキュリティ機能定義ファイルの読み込み中にエラーが発生した場合に出力されます。	
9	KDCGW	SOAP メッセージを送信中にエラーが発生した場合に出力されます。	SOAP 通信基盤が提供する C4Fault クラスの faultString 要素として，ユーザー作成プログラムのメッセージに出力されます。メッセージの本文は，SOAP 通信基盤のトレースファイルにも出力されます。

注

コマンドの詳細については，「4.　Web サービスセキュリティ機能が提供するコマンド」を参照してください。トレースファイルについては，「6.1　トレースを収集する」を参照してください。

SOAP アプリケーション開発支援機能のクラスやトレースファイルについては，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

(2) メッセージ種別

メッセージ種別には次の種類があります。

-E

エラーが発生した場合に出力されるメッセージであることを示します。このメッセージ種別を持つメッセージが出力された場合の対処方法については，「7.2　メッセージの内容」を参照してください。

-I

処理が終了したことを通知するメッセージであることを示します。このメッセージ種別を持つメッセージが出力された場合，処理は正常に終了しているので，対処は必要ありません。

-W

警告を通知するメッセージであることを示します。このメッセージ種別を持つメッセージが出力された場合の対処方法については，「7.2　メッセージの内容」を参照してください。

7.2 メッセージの内容

Web サービスセキュリティ機能のメッセージは，英文で出力されます。この節では，出力されたメッセージの意味，メッセージが出力された要因，およびメッセージが出力された場合の対処方法を，次に示す形式で説明します。

メッセージ ID

メッセージ本文

意味

英文のメッセージの意味を説明します。

要因

メッセージが出力された要因を説明します。

対処

メッセージが出力された場合の対処方法を説明します。

なお，メッセージは，メッセージ ID 順に並んでいます。

7.2.1 KDCGA で始まるメッセージ

KDCGA で始まるメッセージの意味，要因，および対処について説明します。

KDCGA0001-E

A user name is required.

意味

ユーザー名が必要です。

要因

WSSUsernameToken クラスのユーザー名が指定されていません。

対処

SOAP アプリケーションの設計を見直して，WSSUsernameToken クラスの setUsername メソッドでユーザー名を指定するようにしてください。

KDCGA9000-E

An unexpected error occurred during processing. (details = < **詳細** >)

意味

処理中に予期しないエラーが発生しました。< 詳細 > には，エラー内容の詳細が出力されます。

要因

Web サービスセキュリティ機能が提供する API を実行中に原因不明のエラーが発生

しました。

対処

システム管理者に連絡してください。

7.2.2 KDCGC で始まるメッセージ

KDCGC で始まるメッセージの意味，要因，および対処について説明します。

KDCGC0001-E

A file was not found. (file = **＜ファイル名＞**)

意味

＜ファイル名＞のファイルが見つかりません。

要因

指定されたファイルが見つからないか，指定されたファイルに対するアクセス権限がありません。

対処

指定したファイルが存在するかどうか確認してください。指定したファイルが存在するにもかかわらず，このメッセージが出力される場合は，指定したファイルに対してアクセス権限があるかどうか確認してください。ファイルに対するアクセス権限がない場合は，アクセス権限を設定してください。

KDCGC0002-E

A configuration file contains an error. (file = **＜ファイル名＞** , line = **＜行番号＞** , details = **＜詳細＞**)

意味

定義ファイルの内容が不正です。

＜ファイル名＞，＜行番号＞，および＜詳細＞には，それぞれ次の内容が出力されます。

＜ファイル名＞

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルのフルパスが出力されます。

＜行番号＞

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルの中で，定義ファイル構文チェックコマンドによってエラーと判定された行の行番号が出力されます。

＜詳細＞

エラーの詳細が出力されます。＜詳細＞に出力される内容については，マニュアル「Cosminexus XML Processor　ユーザーズガイド」を参照してください。

要因

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルに対して，定義ファイル構文チェックコマンドを実行した結果，エラーが発生しました。

対処

< 詳細 > に出力された内容に従って，Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルを修正してから，定義ファイル構文チェックコマンドを実行してください。

注意事項

SOAP アプリケーションの実行時に，このメッセージが出力された場合は，< ファイル名 >，< 行番号 >，および < 詳細 > は出力されません。ただし，トレースファイルには < ファイル名 >，< 行番号 >，および < 詳細 > が出力されます。トレースファイルについては，「6.1　トレースを収集する」を参照してください。

KDCGC0003-W

The specified attribute contains an invalid value. The default value was assumed. tag name = < **要素名** >, attribute name = < **属性名** > , specified value = < **指定値** > , value to be used = < **仮定する値** >

意味

属性値の内容が正しくないため，デフォルト値を仮定して処理を続行します。< 要素名 >,< 属性名 >,< 指定値 >,< 仮定する値 > にはそれぞれ次の内容が出力されます。

- < 要素名 >
  正しくない値を指定した属性を持つ要素の名称が出力されます。
- < 属性名 >
  正しくない値を指定した属性名が出力されます。
- < 指定値 >
  < 属性名 > に指定した属性値が出力されます。
- < 仮定する値 >
  < 属性名 > で指定した属性のデフォルト値が出力されます。

要因

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルで指定した属性値が正しくないため，デフォルト値を仮定して処理を続行します。

対処

< 属性値 > に出力される値を正しい値にしてください。

KDCGC0004-W

The property contains an invalid value. The default value was assumed. key = < **キー名称** >, specified value = < **指定値** > , value to be used = < **仮定する値** >

意味

指定値の内容が正しくないため，デフォルト値を仮定して処理を続行します。
< キー名称 >,< 指定値 >,< 仮定する値 > にはそれぞれ次の内容が出力されます。

- < キー名称 >
  正しくない値を指定したキー名称が出力されます。
- < 指定値 >
  < キー名称 > に指定した値が出力されます。
- < 仮定する値 >
  < キー名称 > で指定したキーのデフォルト値が出力されます。

要因

環境設定ファイルで指定した指定値が正しくないため，デフォルト値を仮定して処理を続行します。

対処

< キー名称 > に出力されるキーの指定値を正しい値にしてください。

注意事項

このメッセージはトレースファイルだけに出力されます。

7.2.3　KDCGF で始まるメッセージ

KDCGF で始まるメッセージの意味，要因，および対処について説明します。KDCGF で始まるメッセージは，SOAPFault 形式で出力されます。SOAPFault 形式のメッセージには，次に示す四つの項目があります。

FaultCode

Fault コードが出力されます。Fault コードは，接頭辞とローカル部で構成されます。KDCGF で始まるメッセージの Fault コードの接頭辞には，「{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}」が出力されます。ローカル部には，エラーの要因を示す文字列が出力されます。
Fault コードの値は，SOAP 通信基盤の API を使用して取得できます。SOAP 通信基盤が提供する API の仕様については，マニュアル「Cosminexus　SOAP アプリケーション開発ガイド」を参照してください。

FaultString

メッセージ ID およびメッセージの本文が出力されます。メッセージ ID の見方については，「7.1　メッセージの形式」を参照してください。

FaultActor

Fault の生成者が出力されます。

FaultDetails

Fault の詳細が出力されます。

KDCGF0001-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}UnsupportedSecurityToken

FaultString：KDCGF0001-E An unsupported security token was specified. (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞でサポートされていないセキュリティトークン要素が使用されています。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

次のうちのどれかがエラーの要因と考えられます。

- BinarySecurityToken 要素の EncodingType 属性が指定されているにもかかわらず，属性値が「http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary」ではない。
- BinarySecurityToken 要素の ValueType 属性が指定されているにもかかわらず，属性値が「http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3」ではない。
- KeyIdentifier 要素に EncodingType 属性が指定されているにもかかわらず，属性値が「http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary」ではない。
- KeyIdentifier 要素に ValueType 属性が指定されているにもかかわらず，属性値が「http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509SubjectKeyIdentifier」ではない。
- WS-Security の Reference 要素に ValueType 属性が指定されているにもかかわらず，属性値が「http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3」ではない。
- WS-Security の SecurityTokenReference 要素の子要素に，Reference 要素または KeyIdentifier 要素以外の要素が指定されている。
- WS-Security の Security 要素の子要素に，XML 暗号の Reference 要素が指定されている。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。

KDCGF0002-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}UnsupportedAlgorithm

FaultString：KDCGF0002-E An unsupported signature or encryption algorithm was specified. (location = ＜発生場所＞)

FaultActor：なし

FaultDetails：なし

意味

＜発生場所＞でサポートされていない署名アルゴリズム，または暗号アルゴリズムが使用されています。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

次のうちのどれかがエラーの要因と考えられます。

- Canonicalization 要素の Algorithm 属性にサポートされていないアルゴリズムが指定されている。
- SignatureMethod 要素の Algorithm 属性にサポートされていないアルゴリズムが指定されている。
- Transform 要素の Algorithm 属性にサポートされていないアルゴリズムが指定されている。
- Canonicalization 要素の Algorithm 属性に Web サービスセキュリティポリシー定義ファイルで設定されていないアルゴリズムが指定されている。
- SignatureMethod 要素の Algorithm 属性に Web サービスセキュリティポリシー定義ファイルに設定されていないアルゴリズムが指定されている。
- Transform 要素の Algorithm 属性に Web サービスセキュリティポリシー定義ファイルで設定されていないアルゴリズムが指定されている。
- XML 暗号の EncryptionMethod 要素の Algorithm 属性にサポートされていないアルゴリズムが指定されている。
- XML 暗号の EncryptionMethod 要素の Algorithm 属性に Web サービスセキュリティポリシー定義ファイルで設定されていないアルゴリズムが指定されている。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。または，Web サービスセキュリティポリシー定義ファイルの設定を見直してください。

Web サービスセキュリティ機能がサポートしているアルゴリズムについては，「付録 A　標準仕様への対応」を参照してください。Web サービスセキュリティポリシー定義ファイルの設定については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGF0003-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}InvalidSecurity

FaultString：KDCGF0003-E An error occurred during security header processing. (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞のセキュリティヘッダ内でエラーが発生しました。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

次のうちのどれかがエラーの要因と考えられます。

- Web サービスセキュリティポリシー定義ファイルの Timestamp 要素で Created 要素を要求する指定がされているにもかかわらず，受信した SOAP メッセージに Created 要素がない。
- Web サービスセキュリティポリシー定義ファイルの Timestamp 要素で Expires 要素を要求する指定がされているにもかかわらず，受信した SOAP メッセージに Expires 要素がない。
- 受信した SOAP メッセージの Created 要素および Expires 要素の ValueType 属性が xsd:dateTime と異なる。
- 値が必要な要素（Created 要素，Expires 要素，BinarySecurityToken 要素，KeyIdentifier 要素）に値がない。
- Web サービスセキュリティポリシー定義ファイルで BinarySecurityToken 要素を要求する指定がされているにもかかわらず，受信した SOAP メッセージに BinarySecurityToken 要素がない。
- Reference 要素に URI 属性が付与されていない。
- Reference 要素の URI 属性に値が設定されていない。
- Web サービスセキュリティポリシー定義ファイルで SOAP ボディに署名を要求する指定がされているにもかかわらず，受信した SOAP メッセージの SOAP ボディに署名がない。
- 暗号化された SOAP メッセージに KeyInfo 要素がない。
- 暗号化された SOAP メッセージの KeyName 要素で示された鍵が Web サービスセキュリティ機能定義ファイルに記述されていない。
- Web サービスセキュリティポリシー定義ファイルで SOAP ボディの要素の暗号化を要求する指定がされているにもかかわらず，受信した SOAP メッセージの SOAP ボディの要素が暗号化されていない。

- 受信した SOAP メッセージに同じ属性値を持つ Id 属性がある。
- Web サービスセキュリティポリシー定義ファイルの ReceiverPortConfig 要素の Name 属性と My_role 属性に対応した Web サービスセキュリティ機能定義ファイルの設定がありません。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。または，Web サービスセキュリティポリシー定義ファイルの設定を見直してください。
Web サービスセキュリティ機能がサポートしているアルゴリズムについては，「付録 A　標準仕様への対応」を参照してください。Web サービスセキュリティポリシー定義ファイルの設定については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGF0004-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}InvalidSecurityToken

FaultString：KDCGF0004-E An invalid security token was specified (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞で不正なセキュリティトークンが使用されています。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

次のうちのどちらかがエラーの要因と考えられます。

- BinarySecurityToken 要素の ValueType 属性が付与されていない。
- 受信した SOAP メッセージの BinarySecurityToken 要素を，Web サービスセキュリティポリシー定義ファイルに記述された証明書ファイルで検証した場合に，常に検証が失敗する。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。または，Web サービスセキュリティポリシー定義ファイルの設定を見直してください。
Web サービスセキュリティ機能がサポートしているアルゴリズムについては，「付録 A　標準仕様への対応」を参照してください。Web サービスセキュリティポリシー定義ファイルの設定については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGF0005-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}FailedAuthentication

FaultString：KDCGF0005-E A security token could not be authenticated or authorized. (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞のセキュリティトークンは，認証または認可できません。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

KDCGJ0001-E の要因を参照してください。

対処

KDCGJ0001-E の対処を参照してください。

KDCGF0006-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}FailedCheck

FaultString：KDCGF0006-E A signature or decryption was invalid. (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞の署名または復号化が不正です。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

次のうちのどちらかがエラーの要因と考えられます。

- 受信した SOAP メッセージに不正な署名が付与されている。
- 受信した SOAP メッセージが不正に暗号化されている。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。

KDCGF0007-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}SecurityTokenUnavailable

FaultString：KDCGF0007-E A referenced security token cannot be found. (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞で受信した SOAP メッセージの中で，参照先で示されるセキュリティトークン要素が見つかりませんでした。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

- WS-Security の Reference 要素で指定されている BinarySecurityToken 要素が見つかりません。
- WS-Security の KeyIdentifier 要素で指定されているサブジェクトキー識別子を持つ X.509 証明書が Web サービスセキュリティ機能定義ファイルの VerificationKeyStore 要素で指定しているキーストアファイルの中から見つかりません。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。

KDCGF0008-E

FaultCode：{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}MessageExpired

FaultString：KDCGF0008-E An old message or message with an expired date was received. (location = **＜発生場所＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

＜発生場所＞で受信した SOAP メッセージが古いか，メッセージの有効期限が切れています。＜発生場所＞には次の内容が出力されます。

- Server：サーバ側で受信したメッセージでエラーが発生した場合
- Client：クライアント側で受信したメッセージでエラーが発生した場合

要因

次のうちのどちらかがエラーの要因と考えられます。

- 受信した SOAP メッセージの Created 要素の値が古い。
- 受信した SOAP メッセージの Expires 要素で指定されている有効期限が過ぎている。

対処

「要因」に示した内容に該当する SOAP メッセージを送信していないかどうか，メッセージの送信者に確認してください。または，Web サービスセキュリティポリシー定義ファイルの設定を見直してください。
Web サービスセキュリティポリシー定義ファイルの設定については，「付録 C.2 Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGF0009-E

FaultCode : {http://schemas.xmlsoap.org/soap/envelope/}MustUnderstand

FaultString : KDCGF0009-E Can not understand "MustUnderstand" header.(Header name = < **ヘッダ名** >, reason= < **理由** >)

FaultActor : **なし**

FaultDetail : **なし**

意味

mustUnderstand 属性が付加されたヘッダ要素を解釈できません。
＜ヘッダ名＞および＜理由＞には，それぞれ次の内容が出力されます。

- ＜ヘッダ名＞
  mustUnderstand 属性が付加されたヘッダの名称が出力されます。
- ＜理由＞
  解釈できない理由が出力されます。

要因

次の要因が考えられます。

- アプリケーションが Cosminexus Web Services - Security - バージョン 0760 を実装している場合
  次の Security 要素を受信しました。
  ・mustUnderstand 属性が「true」である。
  ・名前空間の値が Web Services Security: SOAP Message Security Working Draft 17 である。
- アプリケーションが Cosminexus Web Services - Security - 旧バージョンを実装している場合
  次の Security 要素を受信しました。
  ・mustUnderstand 属性が「true」である。
  ・名前空間の値が Web Services Security: SOAP Message Security1.1 である。

対処

- アプリケーションが Cosminexus Web Services - Security - バージョン 0760 を実装している場合
  受信する Security 要素の名前空間が Web Services Security: SOAP Message Security1.1 の名前空間になるように，メッセージ送信者に変更を依頼してください。
- アプリケーションが Cosminexus Web Services - Security - 旧バージョンを実装している場合
  受信する Security 要素の名前空間が Web Services Security: SOAP Message Security Working Draft17 の名前空間になるように，メッセージ送信者に変更を依頼してください。

KDCGF0010-E

FaultCode:{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd} InvalidSecurity

FaultString : KDCGF0010-E The namespace < **名前空間** > of the Security header is illegal.

FaultActor : **なし**

FaultDetail : **なし**

意味

セキュリティヘッダの名前空間が不正です。＜名前空間＞には，次の内容が出力されます。

- ＜名前空間＞
  不正な名前空間の名称が出力されます。

要因

次の要因が考えられます。

- アプリケーションが Cosminexus Web Services - Security - バージョン 0760 を実装している場合
  次の Security 要素を受信しました。
  ・mustUnderstand 属性が「false」である。
  ・名前空間の値が Web Services Security: SOAP Message Security Working Draft 17 である。
- アプリケーションが Cosminexus Web Services - Security - 旧バージョンを実装している場合
  次の Security 要素を受信しました。
  ・mustUnderstand 属性が「false」である。
  ・名前空間の値が Web Services Security: SOAP Message Security1.1 である。

対処

- アプリケーションが Cosminexus Web Services - Security - バージョン 0760 を実装している場合

受信する Security 要素の名前空間が Web Services Security: SOAP Message Security1.1 の名前空間になるように，メッセージ送信者に変更を依頼してください。

- アプリケーションが Cosminexus Web Services - Security - 旧バージョンを実装している場合
  受信する Security 要素の名前空間が Web Services Security: SOAP Message Security Working Draft17 の名前空間になるように，メッセージ送信者に変更を依頼してください。

7.2.4 KDCGJ で始まるメッセージ

KDCGJ で始まるメッセージの意味，要因，および対処について説明します。

KDCGJ0001-E

An error occurred during JAAS authentication. (details = ＜詳細＞)

意味

JAAS 認証でエラーが発生しました。

要因

次のうちのどれかがエラーの要因と考えられます。

- Web サービスセキュリティ機能定義ファイルの Username 要素および Password 要素の指定内容が間違っている。
- Web サービスセキュリティ機能定義ファイルの Password 要素の Type 属性の指定値が間違っている。
- LoginModule.login() メソッドの中の実装が間違っている。
- LoginModule.login() メソッドで LoginException をスローしている。
- LoginModule.login() メソッドの戻り値を「false」にしている。※
- JAAS ログインモジュールに必要なログイン構成ファイルがない。
- Cosminexus Component Container のユーザー定義ファイル（usrconf.properties）で指定した場所に JAAS ログインモジュールに必要なログイン構成ファイルがない。
- Cosminexus Component Container のユーザー定義ファイル（usrconf.properties）に JAAS ログインモジュールに必要なログイン構成ファイルが指定されていない。
- ログイン構成ファイルの構文が間違っている。
- ログイン構成ファイルのインデックス値と Web サービスセキュリティ機能定義ファイルで指定したインデックス値が異なる。
- WSSUsernameToken クラスのコンストラクタや setUsername メソッドなどで指定したユーザー名が間違っている。
- WSSUsernameToken クラスの setPassword メソッドで指定したパスワード値が間違っている。

- WSSUsernameToken クラスの setPasswordType メソッドで指定したパスワード形式が間違っている。

注※

LoginModule.login() メソッドで LoginException をスローしている場合，LoginException 生成時に詳細メッセージを指定いるときは，< 詳細 > に詳細なエラー要因が出力されます。

対処

- Web サービスセキュリティ機能定義ファイルを見直して，「要因」に示した間違いがないかどうかを確認してください。
- JAAS ログインモジュールの処理が正しいかどうか，JAAS ログインモジュールの設計を見直してください。
- JAAS ログインモジュールに必要なログイン構成ファイルの設定や格納場所などを見直してください。
- WSSUsernameToken クラスのメソッドの使用方法が正しいかどうか，SOAP アプリケーションの設計を見直してください。

Web サービスセキュリティ機能定義ファイルの要素や属性については，「付録 C.1 Web サービスセキュリティ機能定義ファイルの項目」を参照してください。
WSSUsernameToken クラスのメソッドの使用方法については，「5.4 WSSUsernameToken クラス（UsernameToken 要素の操作）」を参照してください。JAAS ログインモジュールについては，「3.4　認証機能を設定する」を参照してください。ログイン構成ファイルについては，「3.9.4　JAAS ログインモジュールの実装時の注意」を参照してください。
なお，SOAP アプリケーションの設計の詳細については，マニュアル「Cosminexus SOAP アプリケーション開発ガイド」を参照してください。

7.2.5　KDCGK で始まるメッセージ

KDCGK で始まるメッセージの意味，要因，および対処について説明します。

KDCGK0001-I

Generation of a secret key has finished. (file = **＜ファイル名＞**)

意味

共通鍵の生成が完了しました。
< ファイル名 > には，生成した共通鍵のファイル名が出力されます。

KDCGK0010-E

An argument is not specified. (argument = **＜引数＞**)

意味

引数が指定されていません。

＜引数＞には，指定する必要がある引数の名称が出力されます。

要因

共通鍵生成コマンドで指定する必要がある引数が指定されていません。

対処

引数を指定してから，再度，共通鍵生成コマンドを実行してください。
共通鍵生成コマンドで指定する引数については，「4.1　共通鍵生成コマンド（CWSSCreateSecretKey）」を参照してください。

KDCGK0011-E

An invalid argument is specified. (argument = **＜引数＞**)

意味

不正な引数が指定されています。
＜引数＞には，不正と判定された引数の名称が出力されます。

要因

共通鍵生成コマンドで使用できない，不正な引数が指定されています。

対処

不正と判定された引数を削除してから，再度，共通鍵生成コマンドを実行してください。
共通鍵生成コマンドで指定できる引数については，「4.1　共通鍵生成コマンド（CWSSCreateSecretKey）」を参照してください。

KDCGK0012-E

An invalid argument value is specified. (argument = **＜引数＞** , value = **＜引数値＞**)

意味

引数に不正な値が指定されています。
＜引数＞と＜引数値＞には，それぞれ次の内容が出力されます。

＜引数＞
　引数の名称が出力されます。

＜引数値＞
　引数に指定されている値が出力されます。

要因

共通鍵生成コマンドの引数に不正な値が指定されています。

対処

共通鍵生成コマンドの引数，および引数に指定されている値が正しいかどうかを確認してから，再度，共通鍵生成コマンドを実行してください。
共通鍵生成コマンドの引数に指定できる値については，「4.1　共通鍵生成コマンド（CWSSCreateSecretKey）」を参照してください。

KDCGK0013-E

A specified file already exists. (file = **＜ファイル名＞**)

意味

指定した共通鍵のファイルはすでに存在します。

要因

共通鍵生成コマンドの -o オプションで指定したファイル名のファイルがすでに存在しています。

対処

すでに存在しているファイルとは異なるファイル名を指定して，再度，共通鍵生成コマンドを実行してください。

共通鍵生成コマンドで使用できるオプションについては，「4.1　共通鍵生成コマンド（CWSSCreateSecretKey）」を参照してください。

KDCGK0100-E

An error occurred during output of the key to a file. (details = **＜詳細＞**)

意味

ファイルを出力するときにエラーが発生しました。

＜詳細＞には，エラー内容の詳細が出力されます。

要因

共通鍵生成コマンドで生成した共通鍵をファイルに出力するときにエラーが発生しました。

対処

＜詳細＞に表示されるエラーの要因を解決してから，再度，共通鍵生成コマンドを実行してください。＜詳細＞に表示されたエラーの要因がわからない場合は，システム管理者に連絡してください。

KDCGK0101-E

An error occurred during key creation. (details = **＜詳細＞**)

意味

鍵を生成するときにエラーが発生しました。

＜詳細＞には，エラー内容の詳細が出力されます。

要因

共通鍵生成コマンドで共通鍵を生成するときにエラーが発生しました。

対処

＜詳細＞に表示されるエラーの要因を解決してから，再度，共通鍵生成コマンドを実行してください。＜詳細＞に表示されたエラーの要因がわからない場合は，システム管理者に連絡してください。

KDCGK9000-E

An unexpected error occurred during processing. (details = ＜詳細＞)

意味

処理中に予期しないエラーが発生しました。
＜詳細＞には，エラー内容の詳細が出力されます。

要因

共通鍵生成コマンドを実行中に原因不明のエラーが発生しました。

対処

システム管理者に連絡してください。

7.2.6 KDCGO で始まるメッセージ

KDCGO で始まるメッセージの意味，要因，および対処について説明します。

KDCGO0001-I

Validation has finished.

意味

定義ファイルの構文チェックが正常に終了しました。

KDCGO0002-E

Validation has failed.

意味

定義ファイルの構文に誤りがあります。

要因

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルの内容に誤りがあります。

対処

このメッセージの直前に出力されるメッセージの内容に従って，Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルを修正してから，再度，定義ファイル構文チェックコマンドを実行してください。

KDCGO0010-E

An argument is not specified. (argument = ＜引数＞)

意味

引数が指定されていません。
＜引数＞には，指定する必要がある引数の名称が出力されます。

要因

定義ファイル構文チェックコマンドで指定する必要がある引数が指定されていませ

ん。

対処

引数を指定してから，再度，定義ファイル構文チェックコマンドを実行してください。

定義ファイル構文チェックコマンドで指定する引数については，「4.2　定義ファイル構文チェックコマンド（CWSSConfCheck）」を参照してください。

KDCGO0011-E

An invalid argument was specified. (argument = **＜引数＞**)

意味

不正な引数が指定されています。

＜引数＞には，不正と判定された引数の名称が出力されます。

要因

定義ファイル構文チェックコマンドで使用できない，不正な引数が指定されています。

対処

不正と判定された引数を削除してから，再度，定義ファイル構文チェックコマンドを実行してください。

定義ファイル構文チェックコマンドで指定できる引数については，「4.2　定義ファイル構文チェックコマンド（CWSSConfCheck）」を参照してください。

KDCGO0012-E

An invalid argument value was specified. (argument = **＜引数＞** , value = **＜引数値＞**)

意味

引数に不正な値が指定されています。

＜引数＞と＜引数値＞には，それぞれ次の内容が出力されます。

＜引数＞

　引数の名称が出力されます。

＜引数値＞

　引数に指定されている値が出力されます。

要因

定義ファイル構文チェックコマンドの引数に不正な値が指定されています。

対処

定義ファイル構文チェックコマンドの引数，および引数に指定されている値が正しいかどうかを確認してから，再度，定義ファイル構文チェックコマンドを実行してください。

定義ファイル構文チェックコマンドの引数に指定できる値については，「4.2　定義ファイル構文チェックコマンド（CWSSConfCheck）」を参照してください。

KDCGO9000-E

An unexpected error occurred during processing. (details = ＜詳細＞)

意味

処理中に予期しないエラーが発生しました。
＜詳細＞には，エラー内容の詳細が出力されます。

要因

定義ファイル構文チェックコマンドを実行中に原因不明のエラーが発生しました。

対処

システム管理者に連絡してください。

7.2.7 KDCGP で始まるメッセージ

KDCGP で始まるメッセージの意味，要因，および対処について説明します。

KDCGP0001-E

A definition is duplicated. (Name = ＜Name 属性値＞, My_role = ＜My_role 属性値＞)

意味

Name 属性および My_role 属性の値が重複して定義されています。

要因

Web サービスセキュリティポリシー定義ファイル内で，Name 属性および My_role 属性に同じ値を指定した ReceiverPortConfig 要素が複数定義されています。

対処

複数ある ReceiverPortConfig 要素の Name 属性値および My_role 属性値が重複しないように，Web サービスセキュリティポリシー定義ファイルを修正してください。

KDCGP0002-E

An error occurred during reading of an X509 certificate. (details = ＜詳細＞)

意味

X.509 証明書を読み込むときにエラーが発生しました。
＜詳細＞には，エラー内容の詳細が出力されます。

要因

Web サービスセキュリティポリシー定義ファイルの AuthorityCertificateFile 要素の Name 属性値に指定した X.509 証明書ファイルの読み込むときに，エラーが発生しました。次のうちのどれかがエラーの要因と考えられます。

- AuthorityCertificateFile 要素の Name 属性で指定した X.509 証明書ファイルが見つからない。
- AuthorityCertificateFile 要素の Name 属性で指定した X.509 証明書ファイルに対するアクセス権限がない。

- AuthorityCertificateFile 要素の Name 属性で指定した X.509 証明書ファイルの形式が間違っている。

対処

「要因」に示した問題点がないかどうか，Web サービスセキュリティポリシー定義ファイルの設定を見直してください。
Web サービスセキュリティポリシー定義ファイルの設定については，「付録 C.2 Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGP1001-E

The root tag name is invalid. (tag name = **＜要素名＞**)

意味

不正なルート要素名が指定されています。
< 要素名 > には，ルート要素の名称が出力されます。

要因

Web サービスセキュリティポリシー定義ファイルに不正なルート要素名が指定されています。

対処

Web サービスセキュリティポリシー定義ファイルのルート要素の名称を「PolicyConfig」に修正してください。

KDCGP1002-E

A tag name is invalid. (parent tag name = **＜親要素名＞**, tag name = **＜要素名＞**)

意味

不正な名称の要素が指定されています。
< 親要素名 > と < 要素名 > には，それぞれ次の内容が出力されます。

< 親要素名 >
　不正な名称が指定されている要素の親要素の名称が出力されます。

< 要素名 >
　不正な名称が指定されている要素の名称が出力されます。

要因

Web サービスセキュリティポリシー定義ファイルの要素のうち，名称が不正な要素があります。

対処

Web サービスセキュリティポリシー定義ファイルを修正してください。
Web サービスセキュリティポリシー定義ファイルで指定できる要素の名称については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGP1003-E

An attribute value is invalid. (tag name = **＜要素名＞**， attribute name = < **属性名** >, attribute value = **＜属性値＞**)

意味

不正な属性が指定されています。
< 要素名 >，< 属性名 >，および < 属性値 > には，それぞれ次の内容が出力されます。

< 要素名 >
不正な属性が指定されている要素の名称が出力されます。

< 属性名 >
不正な値が指定されている属性の名称が出力されます。

< 属性値 >
< 属性名 > で示された属性に指定されている値が出力されます。

要因

Web サービスセキュリティポリシー定義ファイルの要素のうち，属性値の値が不正な要素があります。

対処

Web サービスセキュリティポリシー定義ファイルを修正してください。
Web サービスセキュリティポリシー定義ファイルの属性に指定できる値については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGP1004-E

＜要素名＞ is undefined.

意味

定義されていない要素があります。
< 要素名 > には，定義されていない要素の名称が出力されます。

要因

Web サービスセキュリティポリシー定義ファイルの必須要素のうち，定義されていない要素があります。

対処

Web サービスセキュリティポリシー定義ファイルを修正してください。
Web サービスセキュリティポリシー定義ファイルの属性に指定できる値については，「付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目」を参照してください。

KDCGP9000-E

An unexpected exception occurred. (details = **＜詳細＞**)

意味

予期しない例外が発生しました。
< 詳細 > には，例外の内容の詳細が出力されます。

要因

Web サービスセキュリティポリシー定義ファイルを解析するときにエラーが発生しました。

対処

< 詳細 > に表示される例外の要因を解決してから，再度，処理を実行してください。
< 詳細 > に表示された例外の要因がわからない場合は，システム管理者に連絡してください。

7.2.8 KDCGS で始まるメッセージ

KDCGS で始まるメッセージの意味，要因，および対処について説明します。

KDCGS0001-E

For **＜要素名＞** , specify either **＜子要素名 1 または属性名 1 ＞** or **＜子要素名 2 または属性名 2 ＞** .

意味

< 要素名 > には，＜子要素名 1 または属性名 1 ＞または＜子要素名 2 または属性名 2 ＞を指定してください。
< 要素名 >，< 子要素名 >，および < 属性名 > には，それぞれ次の内容が出力されます。

< 要素名 >
　どちらか一方だけを指定する必要がある子要素または属性が，両方指定されている要素の名称が出力されます。

< 子要素名 >
　子要素の名称が出力されます。

< 属性名 >
　属性の名称が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの要素のうち，子要素または属性の指定が間違っているものがあります。次のうちのどちらかがエラーの要因と考えられます。

- < 子要素名 1> または < 子要素名 2> のどちらか一方だけを指定する必要があるにもかかわらず，両方の子要素が指定されている。
- < 属性名 1> または < 属性名 2> のどちらか一方だけを指定する必要があるにもかかわらず，両方の属性が指定されている。

対処

Web サービスセキュリティ機能定義ファイルを修正して，< 要素名 > の < 子要素名 1> または < 子要素 2> のどちらか一方を指定するか，< 属性名 1> または < 属性名 2> のどちらか一方を指定するようにしてください。
Web サービスセキュリティ機能定義ファイルの要素や属性の指定回数については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0004-E

An attribute value reference is invalid. (tag name = **＜要素名＞**，attribute name = **＜属性名＞**)

意味

属性で指定された参照先が不正です。
< 要素名 > および < 属性名 > には，それぞれ次の内容が出力されます。

< 要素名 >

参照先が不正な属性を持つ要素の名称が出力されます。

< 属性名 >

参照先が不正な属性の名称が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの属性のうち，参照先が間違っているものがあります。次のうちのどちらかがエラーの要因と考えられます。

- IdRef 属性の値が対応する Id 属性の値と一致していない。
- IdRef 属性に異なる要素の Id 属性の値が指定されている。

対処

Web サービスセキュリティ機能定義ファイルを修正して，< 要素名 > の IdRef 属性に正しい値を指定してください。
Web サービスセキュリティ機能定義ファイルの IdRef 属性に指定する値については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0005-E

A URI or TargetId attribute value must start with #. (tag name = **＜要素名＞**, attribute name = **＜属性名＞**)

意味

URI 属性または TargetId 属性の先頭は「#」でなければなりません。
< 要素名 > および < 属性名 > には，それぞれ次の内容が出力されます。

< 要素名 >

指定されている値の先頭が「#」ではない属性を持つ要素の名称が出力されます。

＜属性名＞

指定されている値の先頭が「＃」ではない属性の名称が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの属性のうち，指定されている値の先頭が「＃」ではない属性があります。

対処

Web サービスセキュリティ機能定義ファイルを修正して，＜属性名＞の属性に指定する値の先頭に「#」を付けてください。

Web サービスセキュリティ機能定義ファイルの属性に指定する値については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0007-E

The specified secret key file does not match the KeyType attribute. (secret key file = **＜共通鍵ファイル名＞**, KeyType = **＜アルゴリズム識別子＞**)

意味

共通鍵のファイルと keytype 属性での指定が一致しません。

＜共通鍵ファイル名＞および＜アルゴリズム識別子＞には，それぞれ次の内容が出力されます。

＜共通鍵ファイル名＞

共通鍵のファイル名が出力されます。

＜アルゴリズム識別子＞

keytype 属性で指定されたアルゴリズム識別子が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの SecretKeyFile 要素の keytype 属性に指定したアルゴリズム識別子と，共通鍵ファイルの内容が異なっています。

対処

Web サービスセキュリティ機能定義ファイルを修正して，SecretKeyFile 要素の keytype 属性と共通鍵ファイルの内容が同じになるようにしてください。keytype 属性には，共通鍵生成コマンドの引数で指定したアルゴリズム識別子と同じものを指定する必要があります。

Web サービスセキュリティ機能定義ファイルの SecretKeyFile 要素の keytype 属性に指定する値については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。共通鍵生成コマンドについては，「4.1　共通鍵生成コマンド（CWSSCreateSecretKey)」を参照してください。

KDCGS0008-E

An error occurred during creation of a secret key. (details = **＜詳細＞**)

意味

共通鍵を生成するときにエラーが発生しました。
< 詳細 > には，エラー内容の詳細が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの SecretKeyFile 要素に指定した共通鍵ファイルから共通鍵を生成するときにエラーが発生しました。次のうちのどちらかがエラーの要因と考えられます。

- SecretKeyFile 要素の指定が間違っている。
- 実行環境の設定が間違っている。

対処

< 詳細 > の内容に従って，Web サービスセキュリティ機能定義ファイルの設定または実行環境を見直してください。

KDCGS0009-E

The EmbedId attribute value is duplicated.

意味

EmbedId 属性の値が重複しています。

要因

Web サービスセキュリティ機能定義ファイルの EmbedId 属性に指定した値が重複しています。

対処

Web サービスセキュリティ機能定義ファイルを修正して，EmbedId 属性の値が重複しないようにしてください。
Web サービスセキュリティ機能定義ファイルの EmbedId 属性に指定する値については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0010-E

< 要素名 > does not exist.

意味

要素が存在しません。
< 要素名 > には，見つからなかった要素の名称が出力されます。

要因

SOAP サービスの URL と Web サービスセキュリティ機能定義ファイルの ReceiverPortConfig 要素の Name 属性に指定した値が一致していません。

対処

Web サービスセキュリティ機能定義ファイルの ReceiverPortConfig 要素の Name 属性に指定した値が SOAP サービスの URL と一致しているかどうか確認してください。

Web サービスセキュリティ機能定義ファイルの ReceiverPortConfig 要素の Name 属性に指定する値については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0011-E

Content for **＜要素名＞** is required.

意味

＜要素名＞の内容を指定する必要があります。
＜要素名＞には，内容が指定されていない要素の名称が出力されます。

要因

Web サービスセキュリティ機能定義ファイルのうち，要素の内容を指定する必要があるにもかかわらず，内容が指定されていない要素があります。

対処

Web サービスセキュリティ機能定義ファイルを修正して，＜要素名＞の内容を指定してください。
Web サービスセキュリティ機能定義ファイルで指定する要素については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0012-E

A definition is duplicated. (Name = **＜** Name **属性値＞**)

意味

Name 属性の指定が重複しています。
<Name 属性値＞には，Web サービスセキュリティ機能定義ファイルの SenderPortConfig 要素の Name 属性に指定されている内容が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの SenderPortConfig 要素のうち，Name 属性に同じ値を指定したものが複数あります。

対処

Web サービスセキュリティ機能定義ファイルを修正して，SenderPortConfig 要素の Name 属性の値が重複しないようにください。
Web サービスセキュリティ機能定義ファイルで指定する要素および属性については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0013-E

A definition is duplicated. (Name = **＜** Name **属性値＞** , My_role = **＜** My_role **属性値＞**)

意味

Name 属性と My_role 属性の値が重複しています。＜ Name 属性値＞および＜

My_role 属性値＞には，Web サービスセキュリティ機能定義ファイルの ReceiverPortConfig 要素の各属性に指定されている内容が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの ReceiverPortConfig 要素のうち，Name 属性と My_role 属性に同じ値を指定したものが複数あります。

対処

Web サービスセキュリティ機能定義ファイルを修正して，ReceiverPortConfig 要素の Name 属性と My_role 属性の値が重複しないようにください。
Web サービスセキュリティ機能定義ファイルで指定する要素および属性については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0014-E

An error occurred during reading of a KeyStore. (details = **＜詳細＞**)

意味

キーストアの読み込み中にエラーが発生しました。＜詳細＞には，エラーの詳細な内容が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの KeyStore 要素の File 属性に指定したキーストアファイルの読み込みでエラーが発生しました。指定したキーストアファイルが存在しなかったり，ファイルに対するアクセス権がなかったり，ファイルの形式が間違っているおそれがあります。

対処

＜詳細＞の内容に従って，Web サービスセキュリティ機能定義ファイルの KeyStore 要素の File 属性の指定を見直してください。
Web サービスセキュリティ機能定義ファイルで指定する要素および属性については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS0015-E

A definition is duplicated. (tag name = < **要素名** >)

意味

要素が重複しています。＜要素名＞には，重複している要素の名称が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの中で，1 回しか指定できない＜要素名＞を複数回指定しています。

対処

Web サービスセキュリティ機能定義ファイルを修正して，＜要素名＞を 1 回だけ指

定するようにしてください。
Web サービスセキュリティ機能定義ファイルで指定する要素の指定回数については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS1002-E

A tag name is invalid. (parent tag name = **＜親要素名＞** , tag name = **＜要素名＞**)

意味

不正な名称の要素があります。
< 親要素名 > および < 要素名 > には，それぞれ次の内容が出力されます。

< 親要素名 >
　名称が不正な要素の親要素の名称が出力されます。

< 要素名 >
　名称が不正な要素の名称が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの要素のうち，名称が不正なものがあります。

対処

Web サービスセキュリティ機能定義ファイルの < 要素名 > を修正してください。
Web サービスセキュリティ機能定義ファイルで指定する要素については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS1003-E

An attribute value is invalid. (tag name = **＜要素名＞** , attribute name = **＜属性名＞** , attribute value = **＜属性値＞**)

意味

属性に指定されている値が不正です。
< 要素名 >，< 属性名 >，および < 属性値 > には，それぞれ次の内容が出力されます。

< 要素名 >
　不正な値が指定されている属性を持つ要素の名称が出力されます。

< 属性名 >
　不正な値が指定されている属性の名称が出力されます。

< 属性値 >
　< 属性名 > に指定されている値が出力されます。

要因

Web サービスセキュリティ機能定義ファイルの属性のうち，指定された値が不正なものがあります。

対処

Web サービスセキュリティ機能定義ファイルの < 属性値 > に指定する値を修正してください。

Web サービスセキュリティ機能定義ファイルの属性に指定する値については,「付録 C.1 Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGS9000-E

An unexpected exception occurred. (details = **＜詳細＞**)

意味

予期しない例外が発生しました。

< 詳細 > には，例外の内容の詳細が出力されます。

要因

Web サービスセキュリティ機能定義ファイルを解析するときにエラーが発生しました。

対処

< 詳細 > に表示される例外の要因を解決してから，再度，処理を実行してください。

< 詳細 > に表示された例外の要因がわからない場合は，システム管理者に連絡してください。

7.2.9 KDCGW で始まるメッセージ

KDCGW で始まるメッセージの意味，要因，および対処について説明します。KDCGW で始まるメッセージは，SOAPFault 形式で出力されます。SOAPFault 形式のメッセージには，次に示す四つの項目があります。

FaultCode

Fault コードが出力されます。Fault コードは，接頭辞とローカル部で構成されます。KDCGW で始まるメッセージの Fault コードの接頭辞には,「{http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/faultcode}」が出力されます。ローカル部には，エラーの要因を示す文字列が出力されます。

Fault コードの値は，SOAP 通信基盤の API を使用して取得できます。SOAP 通信基盤が提供する API の仕様については，マニュアル「Cosminexus SOAP アプリケーション開発ガイド」を参照してください。

FaultString

メッセージ ID およびメッセージの本文が出力されます。メッセージ ID の見方については,「7.1 メッセージの形式」を参照してください。

FaultActor

Fault の生成者が出力されます。

FaultDetails

Fault の詳細が出力されます。

KDCGW0001-E

FaultCode：{http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/faultcode}
<Server.ConfigError **または** Client.ConfigError>

FaultString：KDCGW0001-E An error occurred during configuration file initialization. (file = **＜ファイル名＞** , details = **＜詳細＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルを初期化するときにエラーが発生しました。
<Server.ConfigError または Client.ConfigError>，< ファイル名 >，および < 詳細 > には，それぞれ次の内容が出力されます。

<Server.ConfigError または Client.ConfigError>

エラーがサーバ側で発生したのか，クライアント側で発生したのかを示す文字列が出力されます。サーバ側でエラーが発生している場合は「Server.ConfigError」が，クライアント側でエラーが発生している場合は「Client.ConfigError」が出力されます。

< ファイル名 >

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルのファイル名が出力されます。ファイル名については，「3.1　定義ファイルの設定」を参照してください。

< 詳細 >

KDCGC，KDCGP，または KDCGS で始まるメッセージの本文が出力されます。KDCGC，で始まるメッセージについては，「7.2.2　KDCGC で始まるメッセージ」を参照してください。KDCGP で始まるメッセージについては，「7.2.7　KDCGP で始まるメッセージ」を参照してください。KDCGS で始まるメッセージについては，「7.2.8　KDCGS で始まるメッセージ」を参照してください。

要因

Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルが初期化できませんでした。

対処

＜詳細＞の内容に従って，Web サービスセキュリティ機能定義ファイルまたは Web サービスセキュリティポリシー定義ファイルを修正し，再度，処理を実行してください。処理を再度実行する場合は，SOAP アプリケーションも再度デプロイする必

要があります。

KDCGW0002-E

FaultCode：{http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/faultcode} ＜Server.SigningError **または** Client.SigningError ＞

FaultString：KDCGW0002-E An error occurred during message signature processing. (details = ＜**詳細**＞)

FaultActor：**なし**

FaultDetails：**なし**

意味

メッセージの署名を処理するときにエラーが発生しました。
＜ Server.SigningError または Client.SigningError ＞および＜詳細＞には，それぞれ次の内容が出力されます。

＜ Server.SigningError または Client.SigningError ＞

エラーがサーバ側で発生したのか，クライアント側で発生したのかを示す文字列が出力されます。サーバ側でエラーが発生している場合は「Server.SigningError」が，クライアント側でエラーが発生している場合は「Client.SigningError」が出力されます。

＜詳細＞

エラーの詳細が出力されます。

要因

次のうちのどちらかが要因と考えられます。

- Web サービスセキュリティ機能定義ファイルの CanonicalizationMethod 要素，SignatureMethod 要素，または Transform 要素で指定したアルゴリズムが間違っている。
- Web サービスセキュリティ機能定義ファイルの SignatureTarget 要素で指定した署名対象が間違っている。

対処

＜詳細＞に表示されるエラーの要因を解決してから，再度，処理を実行してください。＜詳細＞に表示されたエラーの要因がわからない場合は，システム管理者に連絡してください。
Web サービスセキュリティ機能定義ファイルの要素で指定する内容については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGW0003-E

FaultCode：{http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/faultcode} ＜Server.EncryptingError **または** Client.EncryptingError ＞

FaultString：KDCGW0003-E An error occurred during message encryption. (details = ＜**詳細**＞)

FaultActor：**なし**

FaultDetails：**なし**

意味

メッセージの暗号化を処理するときにエラーが発生しました。
＜ Server.EncryptingError または Client.EncryptingError ＞および＜詳細＞には，それぞれ次の内容が出力されます。

＜ Server.EncryptingError または Client.EncryptingError ＞
エラーがサーバ側で発生したのか，クライアント側で発生したのかを示す文字列が出力されます。サーバ側でエラーが発生している場合は「Server.EncryptingError」が，クライアント側でエラーが発生している場合は「Client.EncryptingError」が出力されます。

＜詳細＞
エラーの詳細が出力されます。

要因

次のうちのどちらかが要因と考えられます。

- Web サービスセキュリティ機能定義ファイルの ContentsEncryption 要素または KeyEncryption 要素の子要素である EncryptionMethod 要素で指定したアルゴリズムが間違っている。
- Web サービスセキュリティ機能定義ファイルの EncryptionTarget 要素で指定した暗号化対象が間違っている。

対処

＜詳細＞に表示されるエラーの要因を解決してから，再度，処理を実行してください。＜詳細＞に表示されたエラーの要因がわからない場合は，システム管理者に連絡してください。
Web サービスセキュリティ機能定義ファイルの要素で指定する内容については，「付録 C.1　Web サービスセキュリティ機能定義ファイルの項目」を参照してください。

KDCGW9000-E

FaultCode：{http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/faultcode} ＜ Server.ConfigError **または** Client.ConfigError ＞

FaultString：KDCGW9000-E An unexpected error occurred during message transmission processing. (details = **＜詳細＞**)

FaultActor：**なし**

FaultDetails：**なし**

意味

メッセージ送信中に予期しないエラーが発生しました。
＜ Server.ConfigError または Client.ConfigError ＞および＜詳細＞には，それぞれ次の内容が出力されます。

＜ Server.ConfigError または Client.ConfigError ＞

エラーがサーバ側で発生したのか，クライアント側で発生したのかを示す文字列が出力されます。サーバ側でエラーが発生している場合は「Server.ConfigError」が，クライアント側でエラーが発生している場合は「Client.ConfigError」が出力されます。

< 詳細 >

エラーの詳細が出力されます。

要因

メッセージ送信中に原因不明のエラーが発生しました。

対処

＜詳細＞の内容に従って，エラーの要因を解決してから，再度メッセージを送信してください。< 詳細 > に表示されたエラーの要因がわからない場合は，システム管理者に連絡してください。再度メッセージを送信する場合は，SOAP アプリケーションも再度デプロイする必要があります。

付録

付録 A　標準仕様への対応

Web サービスセキュリティ機能は，次の仕様に従っています。

- WS-Security 仕様
  - 07-60 以降のバージョン
    Web Services Security: SOAP Message Security 1.1
  - 07-60 より前のバージョン
    Web Services Security: SOAP Message Security Working Draft 17
- XML 署名標準仕様（2002/2/12　W3C 勧告）
- XML 暗号標準仕様（2002/12/10　W3C 勧告）

WS-Security 仕様の詳細については，OASIS のホームページを参照してください。また，XML 署名および XML 暗号の標準仕様の詳細については，W3C のホームページを参照してください。ここでは，各仕様のうち，Web サービスセキュリティ機能がサポートしている範囲を説明します。

注意事項

- 07-60 以降のバージョンでは，Web Services Security: SOAP Message Security 1.1 以外の仕様を実装した他社製品とは接続できない可能性があります。
- 07-60 より前のバージョンでは，Web Services Security: SOAP Message Security Working Draft 17 以外の仕様を実装した他社製品とは接続できない可能性があります。

付録 A.1　WS-Security 仕様のサポート範囲

Web サービスセキュリティ機能がサポートする WS-Security 仕様の範囲を次の表に示します。

表 A-1　WS-Security 仕様のサポート範囲

項番	WS-Security 仕様			推奨レベル	サポートの有無
	大分類	中分類	小分類		
1	SecurityToken	UsernameToken	PasswordText	推奨	○
2			PasswordDigest	推奨	○
3			Nonce	推奨	○
4			Created	任意	○
5		BinarySecurityToken	X.509v3	任意	○
6			X509PKIPathv1	任意	×
7			PKCS7	任意	×
8			Kerberos5TGT	任意	×
9			Kerberos5ST	任意	×
10		SAML Assertion	—	任意	×
11		XrML	—	任意	×
12		XCBF	—	任意	×
13		EncryptedData Token	—	任意	×
14	TokenReference	Direct Reference	—	推奨	○
15		KeyIdentifiers	—	推奨	○
16		Embedded Reference	—	任意	×
17		KeyNames	—	任意	×
18		ds:KeyInfo	—	推奨	○
19		Encrypted Key Reference	—	任意	×
20	Signature	Algorithms	—	推奨	○
21		Signing Messages	—	必須	○
22		Signing Tokens	—	任意	△※
23		Signature Validation	—	必須	○
24		Signature Confirmation	—	任意	×
25	Encryption	xenc:ReferenceList	—	推奨	×
26		xenc:EncryptedKey	—	推奨	○
27		Encrypted Header	—	任意	×
28		Processing Rules	—	必須	○
29	Security TimeStamp	—	—	推奨	○
30	Error Handling	—	—	推奨	○

（凡例）

○：サポートあり
×：サポートなし
－：該当しない

注※

SecurityToken 要素を直接署名できますが，STR Transform を非サポートのため，SecurityTokenReference 要素を署名できません。

付録 A.2　XML 署名標準仕様のサポート範囲

Web サービスセキュリティ機能がサポートする XML 署名標準仕様の範囲を次の表に示します。

表 A-2　XML 署名標準仕様のサポート範囲

項番	役割	項目	推奨レベル	サポートの有無
1	ダイジェスト値計算	SHA1	必須	○
2	符号化	base64（エンコード）	必須	○
3	署名値計算	HMAC-SHA1	必須	×
4		DSAwithSHA1	必須	○
5		RSAwithSHA1	推奨	○
6	正規化処理	Canonical XML（コメント付き）	推奨	○
7		Canonical XML（コメントなし）	必須	○
8		Exclusive Canonical XML（コメント付き）	任意	○
9		Exclusive Canonical XML（コメントなし）	任意※1	○
10	変換処理	Canonical XML（コメント付き）	推奨	○
11		Canonical XML（コメントなし）	必須	○
12		Exclusive Canonical XML（コメント付き）	任意	○
13		Exclusive Canonical XML（コメントなし）	任意※1	○
14		base64（変換アルゴリズム）	必須	×
15		XSLT	任意	×
16		XPath	推奨	×
17		XPath Filter 2.0	任意	×

項番	役割	項目	推奨レベル	サポートの有無
18		Enveloped Signature	必須	×
19		STR Dereference ※2	推奨	×

（凡例）
○：サポートあり
×：サポートなし

注※ 1
WS-Security 仕様では，サポートを推奨しています。

注※ 2
XML 署名標準仕様で規定されている変換処理ではなく，WS-Security 仕様で規定されているものです。

付録 A.3　XML 暗号標準仕様のサポート範囲

Web サービスセキュリティ機能がサポートする XML 暗号標準仕様の範囲を次の表に示します。

表 A-3　XML 暗号標準仕様のサポート範囲

項番	項目	推奨レベル	サポートの有無
1	Triple DES	必須	○
2	AES-128	必須	○
3	AES-192	任意	×
4	AES-256	必須	×
5	RSA-v1.5	必須	×
6	RSA-OAEP	必須	×
7	Diffie-Hellman Key Values	任意	×
8	Diffie-Hellman Key Agreement	任意	×
9	TRIPLEDES 鍵ラッピング	必須	○
10	AES-128 鍵ラッピング（128bit 鍵）	必須	○
11	AES-192 鍵ラッピング	任意	×
12	AES-256 鍵ラッピング（256bit 鍵）	必須	×
13	XML Decryption Transformation	推奨※	×

（凡例）
○：サポートあり
×：サポートなし

注※
　WS-Security 仕様で規定されている推奨レベルです。

付録 B　下位バージョンからの移行手順

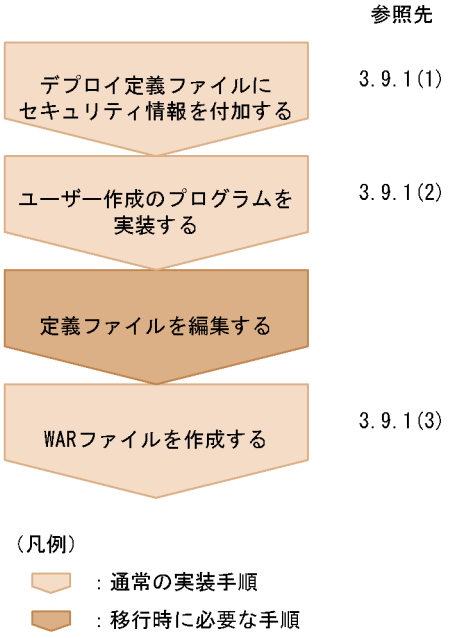
旧バージョンからバージョン 07-60 以降に移行する手順について説明します。旧バージョンで作成したユーザプログラムと 07-60 以降で作成したユーザプログラムは，相互に接続できません。接続した場合は，例外が発生します。なお，旧バージョンで作成したユーザプログラムは，再コンパイルしないで，07-60 以降で実行できます。

(1) サーバ側の移行手順

旧バージョンから 07-60 に移行するには，通常の実装手順に加えて，定義ファイルを編集する必要があります。

Web サービスセキュリティ機能をサーバ側で移行する手順を次の図に示します。

図 B-1　サーバ側の移行手順



ここでは，定義ファイルの編集方法について説明します。定義ファイルの編集以外の手順については，「3.9.1　サーバ側の実装手順」を参照してください。

(a) 定義ファイルを編集する

旧バージョンから 07-60 に移行するには，次の 2 種類の定義ファイルを編集します。

- 機能定義ファイル（security-config.xml）
- ポリシー定義ファイル（policy-config.xml）

定義ファイルの編集手順を次に示します。

1. **各定義ファイルのデフォルト名前空間を次のように変更します。**
   - **機能定義ファイル**

```
http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/securityconfig
```

   - **ポリシー定義ファイル**

```
http://www.hitachi.co.jp/soft/xml/cosminexus/ws/security/0760/policyconfig
```

2. **機能定義ファイルおよびポリシー定義ファイルのプレフィックス** wsse，wsu **に対応する名前空間を次のように変更します。**
   - wsse

```
http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
```

   - wsu

```
http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd
```

3. **各定義ファイルについて，次に示す** XML **要素を変更します。**

   機能定義ファイル
   - BinarySecurityTokenConfig 要素
     BinarySecurityTokenConfig 要素の詳細については，「表 C-12 BinarySecurityTokenConfig」を参照してください。
   - KeyIdentifier 要素
     KeyIdentifier 要素の詳細については，「表 C-24　KeyIdentifier」を参照してください。
   - Password 要素
     Password 要素の詳細については，「表 C-35　Password」を参照してください。

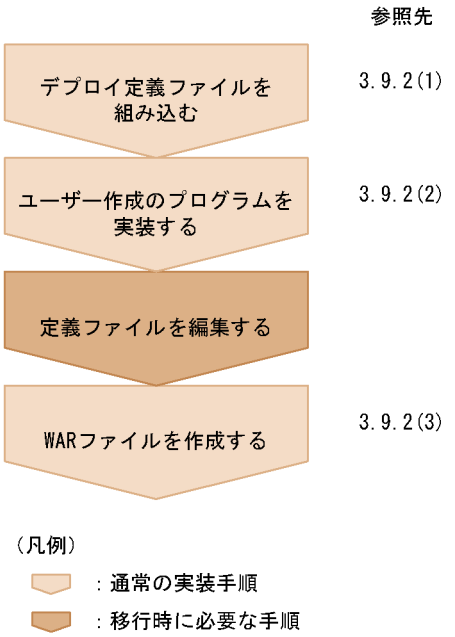
   ポリシー定義ファイル
   - TokenValidation 要素
     TokenValidation 要素の詳細については，「表 C-62　TokenValidation」を参照してください。
   - X509TokenValidation 要素
     X509TokenValidation の詳細については，「表 C-63　X509TokenValidation」を参照してください。

(2) クライアント側が Web アプリケーションの場合の移行手順

旧バージョンから 07-60 に移行するには，通常の実装手順に加えて，定義ファイルを編集する必要があります。

クライアント側が Web アプリケーションの場合に，Web サービスセキュリティ機能を移行する手順を次の図に示します。

図 B-2　クライアント側が Web アプリケーションの場合の移行手順



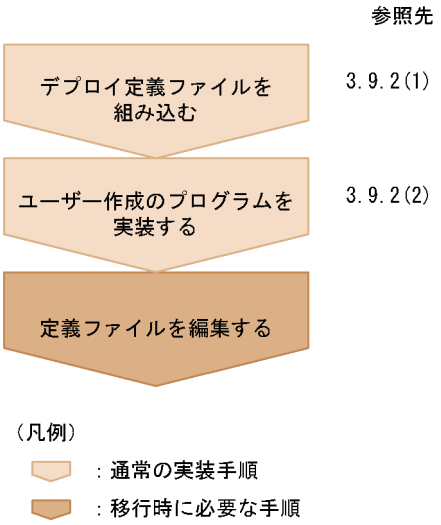
定義ファイルの編集方法については，「付録 B(1)(a)　定義ファイルを編集する」を参照してください。定義ファイルの編集以外の手順については，「3.9.2　クライアント側が Web アプリケーションの場合の実装手順」を参照してください。

(3) クライアント側がコマンドライン Java アプリケーションの場合の移行手順

旧バージョンから 07-60 に移行するには，通常の実装手順に加えて，定義ファイルを編集する必要があります。

クライアント側がコマンドライン Java アプリケーションの場合に，Web サービスセキュリティ機能を移行する手順を次の図に示します。

図 B-3　クライアント側がコマンドライン Java アプリケーションの場合の移行手順



定義ファイルの編集方法については，「付録 B(1)(a)　定義ファイルを編集する」を参照してください。定義ファイルの編集以外の手順については，「3.9.3　クライアント側がコマンドライン Java アプリケーションの場合の実装手順」を参照してください。

付録 C　定義ファイルの項目の詳細

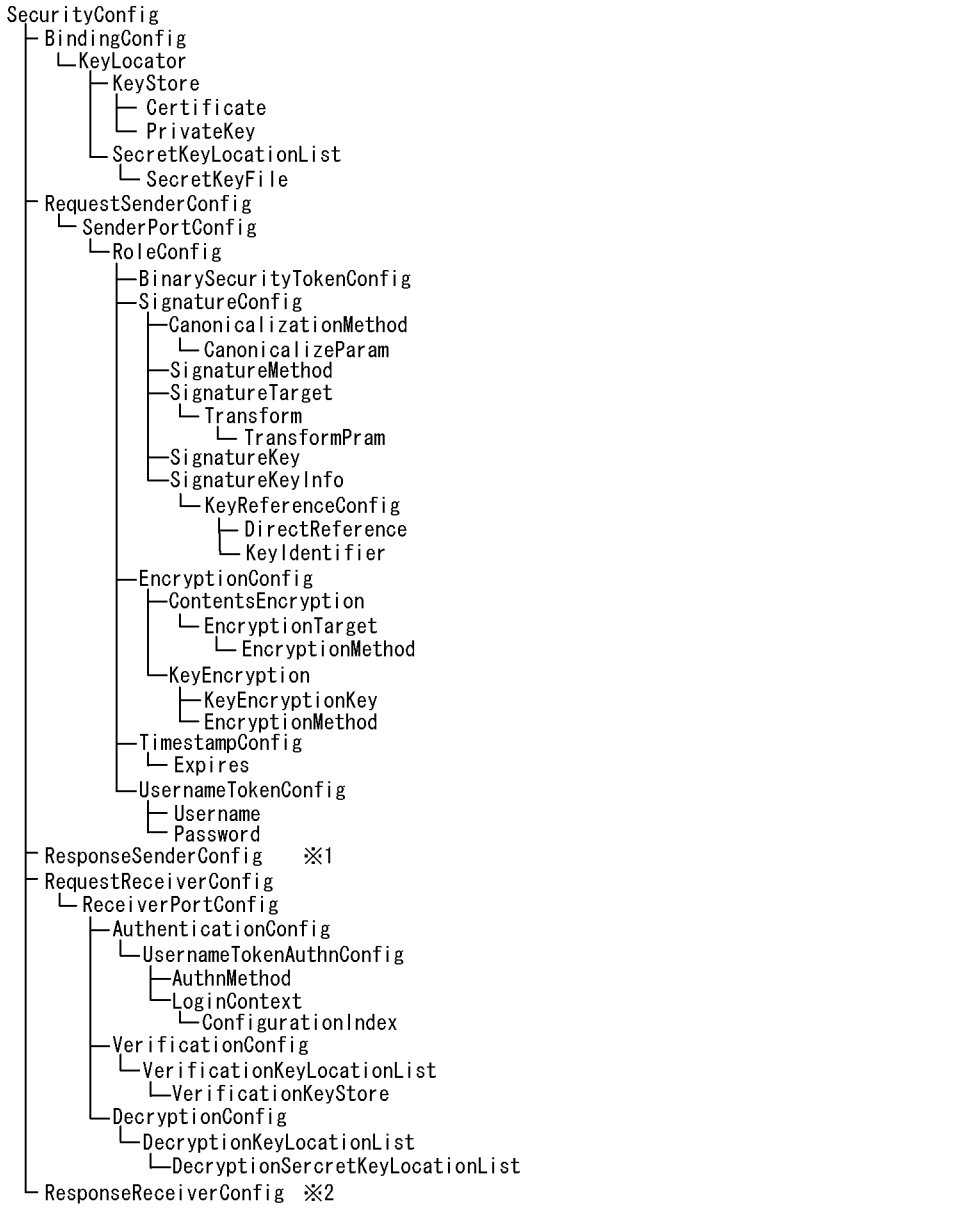
Web サービスセキュリティ機能を使用するには，次の二つの定義ファイルを設定します。

- Web サービスセキュリティ機能定義ファイル
- Web サービスセキュリティポリシー定義ファイル

Web サービスセキュリティの各機能で必要な定義ファイルの項目については，「3.1　定義ファイルの設定」を参照してください。

ここでは，各定義ファイルで設定する要素の指定回数や，注意事項などの詳細を説明します。なお，各定義ファイルの要素の構成は次の図のようになっています。

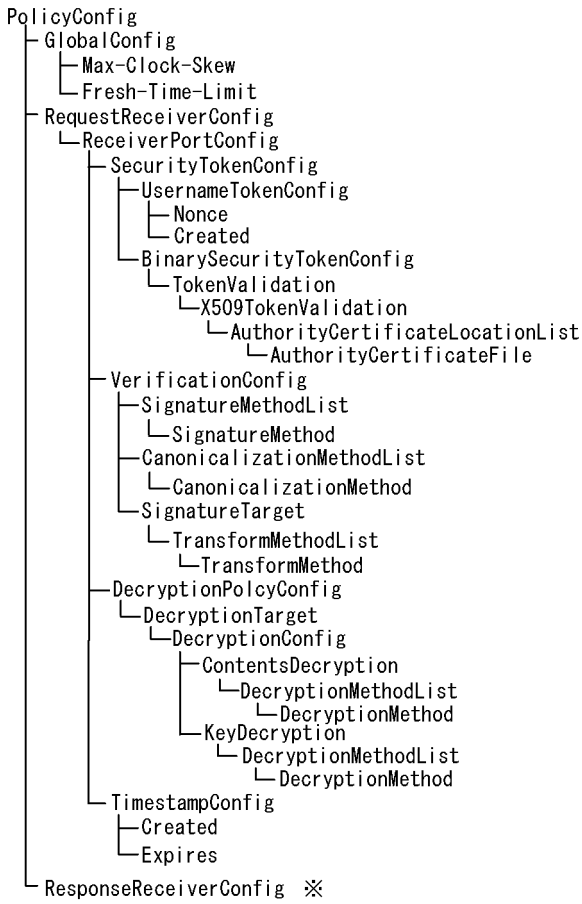
図 C-1　Web サービスセキュリティ機能定義ファイルの要素の構成



注※1
ResponseSenderConfig要素以下の構成は，RequestSenderConfig要素以下の構成と同じです。

注※2
ResponseReceiverConfig要素以下の構成は，RequestReceiverConfig要素以下の構成と同じです。

図 C-2　Web サービスセキュリティポリシー定義ファイルの要素の構成



注※
ResponseReceiverConfig要素以下の構成は，RequestReceiverConfig要素以下の構成と同じです。

付録 C.1　Web サービスセキュリティ機能定義ファイルの項目

Web サービスセキュリティ機能定義ファイルは，XML ファイルです。ここでは，Web サービスセキュリティ機能定義ファイルの要素，および要素で指定できる属性とコンテンツ（子要素）について説明します。なお，表中のデータ型は，XML Schema のデータ型を表しています。

(1)　SecurityConfig

Web サービスセキュリティ定義ファイルのルート要素です。コンテンツ（子要素）は，次の表の順番で指定する必要があります。

表 C-1　SecurityConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	BindingConfig	Web サービスセキュリティの署名，暗号化機能で使用するトークンの情報を指定します。このコンテンツは署名，暗号化機能使用時は必須です。	―	0 または 1
	RequestSenderConfig	リクエストメッセージ送信時の設定を指定します。省略時はリクエストメッセージ送信時に Web サービスセキュリティ機能が実行されません。	―	0 または 1
	ResponseSenderConfig	レスポンスメッセージ送信時の設定を指定します。省略時はレスポンスメッセージ送信時に Web サービスセキュリティ機能が実行されません。	―	0 または 1
	RequestReceiverConfig	リクエストメッセージ受信時の設定を指定します。省略時はリクエストメッセージ受信時に Web サービスセキュリティ機能が実行されません。	―	0 または 1
	ResponseReceiverConfig	レスポンスメッセージ受信時の設定を指定します。省略時はレスポンスメッセージ受信時に Web サービスセキュリティ機能が実行されません。	―	0 または 1

（2） BindingConfig

Web サービスセキュリティの各機能で共通的に使用する情報を指定します。

表 C-2　BindingConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	KeyLocator	鍵の格納場所の情報を指定します。このコンテンツは署名，暗号化機能使用時は必須です。	―	0 または 1

（3） KeyLocator

鍵の格納場所の情報を指定します。コンテンツ（子要素）は，次の表の順番で指定する

必要があります。

表 C-3　KeyLocator

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	KeyStore	Java のキーストア形式の鍵の格納場所情報を指定します。このコンテンツは署名，暗号化機能使用時は必須です。	－	0以上
	SecretKeyLocationList	共通鍵の格納場所の情報を指定します。このコンテンツは暗号化機能使用時は必須です。	－	0 または 1

(4) KeyStore

Java のキーストア形式の鍵の格納場所情報を指定します。

表 C-4　KeyStore

種別	要素	説明	データ型	指定回数
属性	Id	セキュリティ定義ファイル中で一意に識別するための ID 値を指定します。	ID	1
	File	キーストアファイルの名称を " ラベル名 + 拡張子 " の形式で指定します。	String	1
	Type	キーストアのタイプを示す文字列を指定します。	String	1
	Password	キーストアのパスワードを指定します。	String	1
コンテンツ	Certificate	キーストア内の X.509 証明書の情報を指定します。このコンテンツは，キーストア内の証明書を利用する場合は必須です。	－	0以上
	PrivateKey	キーストア内の非公開鍵の情報を指定します。このコンテンツは，キーストア内の非公開鍵を利用する場合は必須です。	－	0以上

(5) Certificate

キーストア内の証明書の情報を指定します。コンテンツはありません。

表 C-5　Certificate

種別	要素	説明	データ型	指定回数
属性	Id	セキュリティ機能定義ファイル中で一意に識別するための ID 値を指定します。	ID	1
	Alias	キーストアファイル中の X.509 証明書のエイリアス名を指定します。	String	1

（6） PrivateKey

キーストア内の非公開鍵の情報を指定します。コンテンツはありません。

表 C-6　PrivateKey

種別	要素	説明	データ型	指定回数
属性	Id	セキュリティ機能定義ファイル中で一意に識別するための ID 値を指定します。	ID	1
	Alias	キーストアファイル中の秘密鍵のエイリアス名を指定します。	String	1
	Password	キーストアファイル中の秘密鍵のパスワードを指定します。	String	1

（7） SecretKeyLocationList

共通鍵の格納場所の情報を指定します。

表 C-7　SecretKeyLocationList

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	SecretKeyFile	共通鍵ファイルの情報を指定します。このコンテンツは暗号化機能使用時は必須です。	―	0以上

（8） SecretKeyFile

共通鍵ファイルの情報を指定します。コンテンツはありません。

表 C-8　SecretKeyFile

種別	要素	説明	データ型	指定回数
属性	Id	セキュリティ定義ファイル中で一意に識別するための ID 値を指定します。	ID	1
	Name	共通鍵作成コマンドで作成した共通鍵ファイル名称を " ラベル名 + 拡張子 " の形式で指定します。	String	1
	KeyType	共通鍵ファイル作成時に指定したアルゴリズム識別子を指定します。	String	1
	KeyName	送信する鍵の識別子を指定します。	String	1

（9） RequestSenderConfig

リクエストメッセージ送信時の設定を指定します。

表 C-9　RequestSenderConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	SenderPortConfig	送信時に，Web サービスセキュリティ機能を適用する SOAP サービスエンドポイントの情報を指定します。	－	1以上

（10）SenderPortConfig

送信時に，Web サービスセキュリティ機能を適用する，SOAP サービスエンドポイントの情報を指定します。ResponseSenderConfig の下に SenderPortConfig を指定する場合は，Name 属性に "*" を指定してください。

表 C-10　SenderPortConfig

種別	要素	説明	データ型	指定回数
属性	Name	Web サービスセキュリティ機能を適用する SOAP サービスの URL を指定します。ここで指定したエンドポイントへメッセージを送信する際に，このコンテンツ以下で指定する Web サービスセキュリティ機能の設定が適用されます。"*" を指定すると，エンドポイントの指定に関係なく Web サービスセキュリティ機能の設定が適用されます。	anyURI	1
コンテンツ	RoleConfig	メッセージ送信時に適用するユーザ名，パスワード，署名，暗号化に関する情報を指定します。省略時はメッセージ送信時に Web サービスセキュリティ機能が実行されません。	－	0以上

（11）RoleConfig

メッセージ送信時に適用するユーザ名，パスワード，署名，暗号化に関する情報を指定します。

表 C-11　RoleConfig

種別	要素	説明	データ型	指定回数
属性	mustUnderstand	送信する SOAP メッセージヘッダの mustUnderstand 属性を "true", "false", "1", "0" のどれかで指定します。この属性省略時は，"true" が仮定されます。"true", "1" を指定した場合は，SOAP メッセージヘッダに mustUnderstand 属性が付加されます。"false", "0" を指定した場合は，SOAP メッセージヘッダに mustUnderstand 属性は付加されません。	boolean	0 または 1
	role	SOAP メッセージヘッダの role 属性（SOAP1.1 での actor 属性）を指定します。	anyURI	1

種別	要素	説明	データ型	指定回数
	Operation	SOAP サービスのサービスメソッド名を指定します。メソッドが複数ある場合は，半角スペースで区切って指定します。この要素を省略した場合は，SOAP サービスのすべてのメソッドに対して，このコンテンツ以下で指定する Web サービスセキュリティ機能の設定が適用されます。この指定は，RequestSenderConfig 以下の RoleConfig に指定した場合だけ有効となります。また，呼び出す SOAP サービスがメッセージング形態の場合は，この指定は無視されます。	NMTOKENS	0 または 1
コンテンツ	BinarySecurityTokenConfig	SOAP メッセージに付与するバイナリセキュリティトークンの情報を指定します。省略時は，SOAP メッセージにバイナリセキュリティトークンは付与されません。	－	0以上
	SignatureConfig	署名に関する情報を指定します。省略時は，SOAP メッセージに署名は付与されません。	－	0以上
	EncryptionConfig	暗号化に関する情報を指定します。省略時は，SOAP メッセージが暗号化されません。	－	0以上
	TimestampConfig	Timestamp 要素に関する情報を指定します。省略時は，SOAP メッセージに Timestamp 要素は付与されません。	－	0 または 1
	UsernameTokenConfig	UsernameToken 要素に関する情報を指定します。省略時は，SOAP メッセージに UsernameToken 要素は付与されません。	－	0以上

（12）BinarySecurityTokenConfig

SOAP メッセージに付与するバイナリセキュリティトークンの情報を指定します。コンテンツはありません。

表 C-12　BinarySecurityTokenConfig

種別	要素	説明	データ型	指定回数
属性	Id	セキュリティ機能定義ファイル中で一意に識別するための ID 値を指定します。	ID	1
	IdRef	バイナリセキュリティトークンとして使用するリソースの位置（Id）を指定します。Certificate 要素の Id 値を指定します。	IDREF	1
	EmbedId	バイナリセキュリティトークン要素をセキュリティヘッダに埋め込む時に付加する Id 値を指定します。	String	1
	EncodingType	セキュリティトークンを送信する際のエンコード種別を指定します。指定できる値は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" だけです。省略時は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" が仮定されます。	anyURI	0 または 1
	ValueType	バイナリセキュリティトークンの種別を指定します。指定できる値は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" だけです。省略時は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" が仮定されます。	anyURI	0 または 1

（13）SignatureConfig

署名に関する情報を指定します。コンテンツ（子要素）は，次の表の順番で指定する必要があります。

表 C-13　SignatureConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―

種別	要素	説明	データ型	指定回数
コンテンツ	CanonicalizationMethod	正規化アルゴリズムを指定します。	―	1
	SignatureMethod	署名アルゴリズムを指定します。	―	1
	SignatureTarget	署名対象を指定します。	―	1以上
	SignatureKey	署名に使用する鍵の情報を指定します。	―	1
	SignatureKeyInfo	署名に使用する鍵の参照情報を指定します。	―	1

（14）CanonicalizationMethod

正規化アルゴリズムを指定します。

表 C-14　CanonicalizationMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	正規化アルゴリズムのアルゴリズム識別子を指定します。	anyURI	1
コンテンツ	CanonicalizeParam	正規化処理で使用するパラメタを指定します。省略時は正規化処理時にパラメタは付与されません。	―	0 または 1

（15）CanonicalizeParam

正規化処理で使用するパラメタを指定します。コンテンツはありません。

表 C-15　CanonicalizeParam

種別	要素	説明	データ型	指定回数
属性	InclusiveNamespaces	CanonicalizationMethod の Algorithm 属性が Exclusive Canonical XML の場合に，Exclusive XML Canonicalization Version 1.0 で規定されている名前空間プレフィクスを指定します。指定方法は，対象となる名前空間プレフィクス名をスペースで区切って指定します（例 "ns1 ns2 ns3"）。CanonicalizationMethod の Algorithm 属性が Exclusive Canonical XML（Exclusive Canonical XML with Comments または Exclusive Canonical XML omits comments）でない場合はこの指定は無視されます。	NMTOKENS	1

（16）SignatureMethod

署名アルゴリズムを指定します。コンテンツはありません。

表 C-16　SignatureMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	署名アルゴリズムのアルゴリズム識別子を指定します。	anyURI	1

（17）SignatureTarget

署名対象を指定します。属性の指定は，Part または TargetId のどちらかが必須になります。

表 C-17　SignatureTarget

種別	要素	説明	データ型	指定回数
属性	Part	署名対象となる SOAP エンベロープ中のエレメントを指定します。指定可能な値は "Body" だけです。	enum	0 または 1
	TargetId	署名対象の Id 値（SOAP メッセージ内の要素にあらかじめ設定済みの wsu:Id の値）を指定します。この指定はメッセージング形態の場合だけ有効です。	String	0 または 1

種別	要素	説明	データ型	指定回数
コンテンツ	Transform	トランスフォームアルゴリズムを指定します（必須）。	―	1以上

（18）Transform

トランスフォームアルゴリズムを指定します。

表 C-18　Transform

種別	要素	説明	データ型	指定回数
属性	Algorithm	トランスフォームアルゴリズムのアルゴリズム識別子を指定します。	anyURI	1
コンテンツ	TransformParam	トランスフォームアルゴリズムで使用するパラメタを指定します。省略時はトランスフォーム処理時にパラメタは付与されません。	―	0 または 1

（19）TransformParam

トランスフォームアルゴリズムで使用するパラメタを指定します。コンテンツはありません。

表 C-19　TransformParam

種別	要素	説明	データ型	指定回数
属性	InclusiveNamespaces	Transform の Algorithm 属性が Exclusive Canonical XML の場合に，Exclusive XML Canonicalization Version 1.0 で規定されている名前空間プレフィクスを指定します。指定方法は，対象となる名前空間プレフィクス名をスペースで区切って指定します（例 "ns1 ns2 ns3"）。 Transform の Algorithm 属性が Exclusive Canonical XML（Exclusive Canonical XML with Comments または Exclusive Canonical XML omits comments）でない場合はこの指定は無視されます。	NMTOKENS	1

（20）SignatureKey

署名に使用する鍵の情報を指定します。コンテンツはありません。

表 C-20　SignatureKey

種別	要素	説明	データ型	指定回数
属性	IdRef	鍵の位置（Binding 要素の PrivateKey 要素で指定する Id 属性）を指定します。	IDREF	1

（21）SignatureKeyInfo

署名に使用する鍵の参照情報を指定します。

表 C-21　SignatureKeyInfo

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	KeyReferenceConfig	署名に使用する鍵のリファレンス情報（KeyInfo 要素内の SecurityTokenReference 要素として設定される情報）を指定します。	－	1

（22）KeyReferenceConfig

署名に使用する鍵のリファレンス情報を指定します。コンテンツは，DirectReference または KeyIdentifier のどちらかを指定します。

表 C-22　KeyReferenceConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	DirectReference	鍵の参照メカニズムを DirectReference にする場合の情報を指定します。	－	0 または 1
	KeyIdentifier	鍵の参照メカニズムを KeyIdentifier にする場合の情報を指定します。	－	0 または 1

（23）DirectReference

鍵の参照メカニズムを DirectReference にする場合の情報を指定します。コンテンツはありません。URI 属性は，先頭に "#" を必ず付加してください。

表 C-23　DirectReference

種別	要素	説明	データ型	指定回数
属性	URI	セキュリティヘッダに埋め込む鍵（バイナリセキュリティトークン）の位置（URI）を指定します。ここで指定した値がセキュリティヘッダ内の wsse:Reference 要素の URI 属性に設定されます。ここで指定する値は，BinarySecurityTokenConfig 要素の EmbedId の値を参照する値である必要があります。	anyURI	1

（24）KeyIdentifier

鍵の参照メカニズムを KeyIdentifier にする場合の情報を指定します。コンテンツはありません。

表 C-24　KeyIdentifier

種別	要素	説明	データ型	指定回数
属性	IdRef	鍵として使用するキーストアの非公開鍵の位置（Certificate 要素の Id 値）を指定します。	IDREF	1
	EncodingType	KeyIdentifier 要素のエンコード種別を指定します。 指定できる値は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" だけです。省略時は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary" が仮定されます。	anyURI	0 または 1

種別	要素	説明	データ型	指定回数
	ValueType	KeyIdentifier 要素の種別を指定します。指定できる値は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509SubjectKeyIdentifier" だけです。省略時は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509SubjectKeyIdentifier" が仮定されます。	anyURI	0 または 1

（25）EncryptionConfig

暗号化に関する情報を指定します。

表 C-25　EncryptionConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	ContentsEncryption	メッセージ内容の暗号化に関する情報を指定します。	－	1
	KeyEncryption	鍵の暗号化に関する情報を指定します。EncryptionType が "ContentsEncryption" の場合は不要です。	－	1

（26）ContentsEncryption

メッセージ内容の暗号化に関する情報を指定します。

表 C-26　ContentsEncryption

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	EncryptionTarget	暗号化対象を指定します。	－	1

（27）EncryptionTarget

暗号化対象を指定します。Part または TargetId のどちらかを指定してください。Part および TargetId の両方を指定した場合は，Part の指定だけが有効になります。また，TargetId を指定する場合は，対象となる SOAP メッセージのエレメントに同じ値の wsu:Id 属性が指定されている必要があります。

表 C-27　EncryptionTarget

種別	要素	説明	データ型	指定回数
属性	Part	暗号化対象となる SOAP エンベロープ中のエレメントを指定します。指定可能な値は "BodyContent" だけです。	enum	0 または 1
	TargetId	暗号化対象の Id 値（SOAP メッセージ内の要素にあらかじめ設定済みの wsu:Id の値）を指定します。この指定はメッセージング SOAP サービスの場合だけ有効です。	String	0 または 1
	EmbedId	暗号化適用後のエレメントに付加する Id 値（wsu:Id 属性として設定される）を指定します。この属性を省略した場合は Id 値は Web サービスセキュリティ機能独自の値を自動的に付加します。	String	0 または 1
コンテンツ	EncryptionMethod	暗号化アルゴリズムを指定します。	—	1

（28）EncryptionMethod

暗号化アルゴリズムを指定します。コンテンツはありません。

表 C-28　EncryptionMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	暗号化アルゴリズムのアルゴリズム識別子を指定します。	anyURI	1

（29）KeyEncryption

鍵の暗号化に関する情報を指定します。

表 C-29　KeyEncryption

種別	要素	説明	データ型	指定回数
属性	—	—	—	—
コンテンツ	EncryptionMethod	暗号化アルゴリズムを指定します。	—	1
	KeyEncryptionKey	鍵の暗号化に使用する鍵の情報を指定します。	—	1

（30）KeyEncryptionKey

鍵の暗号化に使用する鍵の情報を指定します。コンテンツはありません。

表 C-30　KeyEncryptionKey

種別	要素	説明	データ型	指定回数
属性	IdRef	使用する鍵の位置（SecretKeyFile 要素で指定する Id 属性）を指定します。	IDREF	1

（31）TimestampConfig

Timestamp 要素に関する情報を指定します。

表 C-31　TimestampConfig

種別	要素	説明	データ型	指定回数
属性	EmbedId	Timestamp 要素をセキュリティヘッダに埋め込む時の Id 値を指定します。省略時は Timestamp 要素をセキュリティヘッダに埋め込む時に Id は付与されません。	String	0 または 1
	Created	Timestamp 要素に Created 要素を付加するかどうかを "true", "false", "1", "0" のどれかで指定します。"true", "1" を指定した場合 Created 要素が付加されます。"false", "0" を指定した場合 Created 要素は付加されません。省略時は "false" が仮定されます。	boolean	0 または 1

種別	要素	説明	データ型	指定回数
	Expires	Timestamp 要素に Expires 要素を付加するかどうかを "true", "false", "1", "0" のどれかで指定します。"true", "1" を指定した場合 Expires 要素が付加されます。"false", "0" を指定した場合 Expires 要素は付加されません。省略時は "false" が仮定されます。	boolean	0 または 1
コンテンツ	Expires	Expires 要素に関する情報を指定します。	—	0 または 1

（32）Expires

Expires 要素に関する情報を指定します。コンテンツはありません。

表 C-32　Expires

種別	要素	説明	データ型	指定回数
属性	Value	有効期限を設定します。現在時刻からの相対時間をミリ秒単位で指定します。指定可能な範囲は 1,000 ～ 2,147,483,647 です。範囲外の値を指定した場合，または省略時は 300,000（5 分）が仮定されます。	int	0 または 1

（33）UsernameTokenConfig

UsernameToken 要素に関する情報を指定します。

表 C-33　UsernameTokenConfig

種別	要素	説明	データ型	指定回数
属性	Id	Web サービスセキュリティ機能定義ファイル中で，UsernameTokenConfig 要素を一意に識別するための ID 値を指定します。	ID	1

種別	要素	説明	データ型	指定回数
	EmbedId	UsernameToken 要素をセキュリティヘッダに埋め込む時の Id 値を指定します。省略時は UsernameToken 要素をセキュリティヘッダに埋め込む時に Id が付与されません。	String	0 または 1
コンテンツ	Username	ユーザ名に関する情報を指定します。	―	1
	Password	パスワードに関する情報を指定します。省略時は UsernameToken 要素にパスワードが付与されません。	―	0 または 1

（34）Username

ユーザ名に関する情報を指定します。

表 C-34　Username

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	ユーザ名	認証に使用するユーザ ID 値を指定します。	String	1

（35）Password

パスワードに関する情報を指定します。

表 C-35　Password

種別	要素	説明	データ型	指定回数
属性	Type	パスワードの形式を指定します。指定できる値は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText"（テキスト），または "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordDigest"（ダイジェスト）の 2 種類だけです。省略時は，"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText" が仮定されます。	anyURI	0 または 1
コンテンツ	パスワード値	認証に使用するパスワード値を指定します。	String	1

（36）ResponseSenderConfig

レスポンスメッセージ送信時の設定を指定します。

表 C-36　ResponseSenderConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	SenderPortConfig	送信時に Web サービスセキュリティ機能を適用する SOAP サービスエンドポイントの情報を指定します。	－	1以上

（37）RequestReceiverConfig

リクエストメッセージ受信時の設定を指定します。

表 C-37　RequestReceiverConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－

種別	要素	説明	データ型	指定回数
コンテンツ	ReceiverPortConfig	Web サービスセキュリティ機能を適用する SOAP サービスエンドポイントの情報を指定します。	―	1以上

（38）ReceiverPortConfig

受信時に Web サービスセキュリティ機能を適用する SOAP サービスエンドポイントの情報を指定します。

表 C-38　ReceiverPortConfig

種別	要素	説明	データ型	指定回数
属性	Name	メッセージを受信する SOAP サービスの URL を指定します。	anyURI	1
	My_role	Web サービスセキュリティ機能を適用する SOAP サービスのロール名を URI 形式で指定します。受信した SOAP メッセージのセキュリティヘッダ中の role 属性値が，ここで指定したロール名と一致した場合に，PortTypeConfig 以下に指定した動作定義に従って Web サービスセキュリティ機能を実行します。この属性を省略した場合は，受信した SOAP メッセージのセキュリティヘッダ中の role 属性の内容（role 属性がない場合も含む）にかかわらず，PortTypeConfig 以下に指定した動作定義に従って Web サービスセキュリティ機能を実行します。	anyURI	1
コンテンツ	AuthenticationConfig	セキュリティトークンの認証に関する情報を指定します。省略時はセキュリティトークンの認証を行いません。	―	0 または 1
	VerificationConfig	署名検証に関する情報を指定します。省略時は署名検証を行いません。	―	0 または 1
	DecryptionConfig	復号化に関する情報を指定します。省略時は復号化を行いません。	―	0 または 1

（39）AuthenticationConfig

セキュリティトークンの認証に関する情報を指定します。

表 C-39　AuthenticationConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	UsernameTokenAuthnConfig	UsernameToken の認証に関する情報を指定します。省略時は UsernameToken の認証を行いません。	－	0 または 1

（40）UsernameTokenAuthnConfig

UsernameToken の認証に関する情報を指定します。LoginContext タグを省略した場合は，UsernameToken の認証をしないで処理を続行します。

表 C-40　UsernameTokenAuthnConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	AuthnMethod	認証方式に関する情報を指定します。	－	1
	LoginContext	JAAS 認証に関する情報を指定します。AuthnMethod に "JAASAuthn" を指定した場合，この要素は必須です。	－	0 または 1

（41）AuthnMethod

認証方式に関する情報を指定します。コンテンツはありません。

表 C-41　AuthnMethod

種別	要素	説明	データ型	指定回数
属性	Type	認証方式を指定します。指定できる値は "JAASAuthn" だけです。	enum	1

（42）LoginContext

JAAS 認証に関する情報を指定します。

表 C-42　LoginContext

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	ConfigurationIndex	ログイン構成のインデックスを指定します。	―	1

（43）ConfigurationIndex

ログイン構成のインデックスを指定します。

表 C-43　ConfigurationIndex

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	インデックス値	LoginContext クラスをインスタンス化する際に使用するログイン構成のインデックス（jaas.conf 内のインデックス名）を指定します。	―	1

（44）VerificationConfig

署名検証に関する情報を指定します。

表 C-44　VerificationConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	VerificationKeyLocationList	署名検証に用いる鍵に関する情報を指定します。	―	1

（45）VerificationKeyLocationList

署名検証に用いる鍵に関する情報を指定します。受信したセキュリティヘッダ内の署名鍵の参照形式（SecurityTokenReference 要素で示されます）が "wsse:Reference" の場

合，この要素の VerificationKeyStore タグで指定された内容は無視されます。

表 C-45　VerificationKeyLocationList

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	VerificationKeyStore	署名検証に用いる鍵の所在に関する情報を指定します。	－	1以上

（46）VerificationKeyStore

署名検証に用いる鍵の所在に関する情報を指定します。コンテンツはありません。

表 C-46　VerificationKeyStore

種別	要素	説明	データ型	指定回数
属性	IdRef	使用する鍵を含む KeyStore 要素の Id 属性を指定します。	IDREF	1

（47）DecryptionConfig

復号化に関する情報を指定します。

表 C-47　DecryptionConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	DecryptionKeyLocationList	復号に用いる鍵の所在に関する情報を指定します。	－	1

（48）DecryptionKeyLocationList

復号に用いる鍵の所在に関する情報を指定します。

表 C-48　DecryptionKeyLocationList

種別	要素	説明	データ型	指定回数
属性	－	－	－	－

種別	要素	説明	データ型	指定回数
コンテンツ	DecryptionSecretKeyLocationList	復号に用いる鍵の所在に関する情報を指定します。	－	1以上

（49）DecryptionSecretKeyLocationList

復号に用いる鍵の所在に関する情報を指定します。コンテンツはありません。

表 C-49　DecryptionSecretKeyLocationList

種別	要素	説明	データ型	指定回数
属性	IdRef	使用する鍵を含む SecretKeyFile 要素の Id 属性の値を指定します。	IDREF	1

（50）ResponseReceiverConfig

リクエストメッセージ受信時の設定を指定します。

表 C-50　ResponseReceiverConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	ReceiverPortConfig	Web サービスセキュリティ機能を適用する SOAP サービスエンドポイントの情報を指定します。	－	1

付録 C.2　Web サービスセキュリティポリシー定義ファイルの項目

Web サービスセキュリティポリシー定義ファイルは，XML ファイルです。ここでは，Web サービスセキュリティポリシー定義ファイルの要素，および要素で指定できる属性とコンテンツ（子要素）について説明します。なお，表中のデータ型は，XML Schema のデータ型を表しています。

（1） PolicyConfig

ポリシー定義ファイルのルート要素です。

表 C-51　PolicyConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	GlobalConfig	Web サービスセキュリティポリシー定義で共通的に使用するポリシーを指定します。省略時は，GlobalConfig 要素内の値はすべてデフォルト値が仮定されます。	―	0 または 1
	RequestReceiverConfig	リクエストメッセージ受信時の設定を指定します。省略時は，リクエストメッセージ受信時のポリシーチェックが行われません。	―	0 または 1
	ResponseReceiverConfig	レスポンスメッセージ受信時の設定を指定します。省略時は，レスポンスメッセージ受信時のポリシーチェックが行われません。	―	0 または 1

(2) GlobalConfig

Web サービスセキュリティポリシー定義で共通的に使用するポリシーを指定します。

表 C-52　GlobalConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	Max-Clock-Skew	有効期限をチェックする際の時間差に関する情報を指定します。省略時は Max-Clock-Skew 要素内の値はすべてデフォルト値が仮定されます。	―	0 または 1
	Fresh-Time-Limit	UsernameToken 要素，Timestamp 要素内の Created 要素の有効期間に関するポリシーチェックの情報を指定します。省略時は Fresh-Time-Limit 要素内の値はすべてデフォルト値が仮定されます。	―	0 または 1

(3) Max-Clock-Skew

有効期限をチェックする際の時間差に関する情報を指定します。コンテンツはありません。

表 C-53　Max-Clock-Skew

種別	要素	説明	データ型	指定回数
属性	Value	UsernameToken 要素および Timestamp 要素の子要素である Created 要素，または Timestamp 要素の子要素である Expires 要素に指定された値に基づいて SOAP メッセージの有効期限を確認する場合に，送信側と受信側との時間の差をどこまで許容するかを指定します。指定するときの単位はミリ秒です。 指定できる範囲は，1 ～ 2,147,483,647 の間です。範囲外の値を指定した場合，または指定を省略した場合は，0 ミリ秒が仮定されます。	int	1

(4) Fresh-Time-Limit

UsernameToken 要素，および Timestamp 要素内の Created 要素の有効期間に関するポリシーチェックの情報を指定します。コンテンツはありません。

表 C-54　Fresh-Time-Limit

種別	要素	説明	データ型	指定回数
属性	Value	UsernameToken 要素および Timestamp 要素の子要素である Created 要素に指定された値を確認する場合に，送信側と受信側との時間の差をどこまで許容するかを指定します。 指定するときの単位はミリ秒です。 指定できる範囲は，1,000 ～ 2,147,483,647 の間です。範囲外の値を指定した場合，または指定を省略した場合は，300,000 ミリ秒が仮定されます。	int	1

(5) RequestReceiverConfig

リクエストメッセージ受信時の設定を指定します。

表 C-55　RequestReceiverConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－

種別	要素	説明	データ型	指定回数
コンテンツ	ReceiverPortConfig	Web サービスセキュリティポリシー定義を適用する SOAP サービスエンドポイントの情報を指定します。	―	1以上

(6) ReceiverPortConfig

Web サービスセキュリティポリシー定義を適用する SOAP サービスエンドポイントの情報を指定します。

表 C-56　ReceiverPortConfig

種別	要素	説明	データ型	指定回数
属性	Name	Web サービスセキュリティポリシー定義を適用する SOAP サービスの URL を指定します。 アスタリスク "*" を指定した場合，すべての SOAP サービスに対して Web サービスセキュリティ定義が適用されます。	anyURI	1
	My_role	Web サービスセキュリティポリシー定義を適用する SOAP サービスのロール名を，URI で指定します。受信した SOAP メッセージのセキュリティヘッダ内で，RoleConfig 要素の role 属性に指定されている値が My_role 属性で指定したロール名と一致した場合に，ReceiverPortConfig 要素で指定した定義に従って Web サービスセキュリティポリシー定義が適用されます。	anyURI	1
コンテンツ	SecurityTokenConfig	セキュリティトークンに関するポリシーチェックの情報を指定します。省略時はセキュリティトークンに関するポリシーチェックが行われません。	―	0 または 1
	VerificationConfig	署名に関するポリシーチェックの情報を指定します。省略時は署名に関するポリシーチェックが行われません。	―	0 または 1
	DecryptionPolicyConfig	復号化に関するポリシーチェックの情報を指定します。省略時は復号化に関するポリシーチェックが行われません。	―	0 または 1

種別	要素	説明	データ型	指定回数
	TimestampConfig	タイムスタンプに関するポリシーチェックの情報を指定します。省略時はタイムスタンプに関するポリシーチェックが行われません。	―	0 または 1

（7） SecurityTokenConfig

セキュリティトークンに関するポリシーチェックの情報を指定します。

表 C-57　SecurityTokenConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	UsernameTokenConfig	UsernameToken 要素に関するポリシーチェックの情報を指定します。省略時は UsernameToken に関するポリシーチェックが行われません。	―	0 または 1
	BinarySecurityTokenConfig	バイナリセキュリティトークンに関するポリシーチェックの情報を指定します。省略時はバイナリセキュリティトークンに関するポリシーチェックが行われません。	―	0 または 1

（8） UsernameTokenConfig

UsernameToken 要素に関するポリシーチェックの情報を指定します。

表 C-58　UsernameTokenConfig

種別	要素	説明	データ型	指定回数
属性	Required	セキュリティヘッダ内の UsernameToken 要素の有無をチェックするかどうかを "true", "false", "1 ", "0" のどれかで指定します。"true", "1" を指定した場合，セキュリティヘッダ内の UsernameToken 要素の有無をチェックし，存在しないときはポリシー違反として SOAP Fault をスローします。"false", "0" を指定した場合，セキュリティヘッダ内の UsernameToken 要素の有無をチェックしません。"false", 0" を指定した場合，UsernameToken 要素が存在しても処理しません。	boolean	1

種別	要素	説明	データ型	指定回数
コンテンツ	Nonce	Nonce 要素に関するポリシーチェックの情報を指定します。省略時は Nonce 要素に関するポリシーチェックが行われません。	—	0 または 1
	Created	UsernameToken 要素内の Created 要素に関するポリシーチェックの情報を指定します。省略時は Created 要素に関するポリシーチェックが行われません。	—	0 または 1

（9） Nonce

Nonce 要素に関するポリシーチェックの情報を指定します。コンテンツはありません。

表 C-59　Nonce

種別	要素	説明	データ型	指定回数
属性	Required	UsernameToken 要素内の Nonce 要素の有無をチェックするかどうかを "true"，"false"，"1 ”，"0" のどれかで指定します。"true"，"1" を指定した場合 UsernameToken 要素内の Nonce 要素の有無をチェックし，存在しない場合はポリシー違反として SOAP Fault をスローします。"false"，"0" を指定した場合 UsernameToken 要素内の Nonce 要素の有無をチェックしません。"false"，"0" を指定した場合，Nonce 要素が存在しても処理しません。	boolean	1

（10）Created

UsernameToken 要素内の Created 要素に関するポリシーチェックの情報を指定します。

表 C-60　Created

種別	要素	説明	データ型	指定回数
属性	Required	UsernameToken 要素内の Created 要素の有無をチェックするかどうかを "true"，"false"，"1 ”，"0" のどれかで指定します。"true"，"1" を指定した場合 UsernameToken 要素内の Created 要素の有無をチェックし，存在しない場合はポリシー違反として SOAP Fault をスローします。"false"，"0" を指定した場合 UsernameToken 要素内の Created 要素の有無をチェックしません。"false"，"0" を指定した場合，Created 要素が存在しても処理しません。	boolean	1

（11）BinarySecurityTokenConfig

バイナリセキュリティトークンに関するポリシーチェックの情報を指定します。

表 C-61　BinarySecurityTokenConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	TokenValidation	BinarySecurityToken 要素の検証に関するポリシーチェックの情報を指定します。	―	1

（12）TokenValidation

BinarySecurityToken 要素の検証に関するポリシーチェックの情報を指定します。

表 C-62　TokenValidation

種別	要素	説明	データ型	指定回数
属性	―	―	―	―

種別	要素	説明	データ型	指定回数
コンテンツ	X509TokenValidation	X.509 証明書の検証に関するポリシーチェックの情報を指定します。この要素が指定されている場合，受信した SOAP メッセージに X.509 証明書の BinarySecurityToken 要素（ValueType が "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" である BinarySecurityToken 要素）が存在しない場合，ポリシー違反として SOAP Fault をスローします。省略時は X.509 証明書の検証に関するポリシーチェックが行われません。	―	0 または 1

（13）X509TokenValidation

X.509 証明書の検証に関するポリシーチェックの情報を指定します。

表 C-63　X509TokenValidation

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	AuthorityCertificateLocationList	X.509 証明書の検証に関するポリシーチェックの情報を指定します。	―	1

（14）AuthorityCertificateLocationList

X.509 証明書の検証に関するポリシーチェックの情報を指定します。

表 C-64　AuthorityCertificateLocationList

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	AuthorityCertificateFile	X.509 証明書の署名検証に使用する証明書ファイルの情報を指定します。	―	1以上

（15）AuthorityCertificateFile

X.509 証明書の署名検証に使用する証明書ファイルの情報を指定します。コンテンツはありません。

表 C-65　AuthorityCertificateFile

種別	要素	説明	データ型	指定回数
属性	Name	X.509 証明書の署名検証に使用する証明書ファイル名を指定します。	String	1

（16）VerificationConfig

署名に関するポリシーチェックの情報を指定します。VerificationConfig 要素の指定があり，署名がされていないメッセージを受信した場合はポリシーチェック違反として SOAP Fault をスローします。

表 C-66　VerificationConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	SignatureMethodList	署名アルゴリズムに関するポリシーチェックの情報を指定します。	－	1
	CanonicalizationMethodList	正規化アルゴリズムに関するポリシーチェックの情報を指定します。	－	1
	SignatureTarget	署名個所に関するポリシーチェックの情報を指定します。	－	1

（17）SignatureMethodList

署名アルゴリズムに関するポリシーチェックの情報を指定します。受信メッセージ中の署名アルゴリズムが，この要素の SignatureMethod タグで指定した，どのアルゴリズムとも一致しない場合は，ポリシーチェック違反として SOAP Fault をスローします。

表 C-67　SignatureMethodList

種別	要素	説明	データ型	指定回数
属性	－	－	－	－

種別	要素	説明	データ型	指定回数
コンテンツ	SignatureMethod	署名アルゴリズムに関する情報を指定します。省略時は署名アルゴリズムに関するポリシーチェックが行われません。	－	0以上

（18）SignatureMethod

署名アルゴリズムに関する情報を指定します。コンテンツはありません。

表 C-68　SignatureMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	受信側で処理可能な署名アルゴリズムの URI を指定します。	anyURI	1

（19）CanonicalizationMethodList

正規化アルゴリズムに関するポリシーチェックの情報を指定します。受信メッセージ中の正規化アルゴリズムが，この要素の CanonicalizationMethod タグで指定した，どのアルゴリズムとも一致しない場合は，ポリシーチェック違反として SOAP Fault をスローします。

表 C-69　CanonicalizationMethodList

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	CanonicalizationMethod	正規化アルゴリズムに関する情報を指定します。省略時は正規化アルゴリズムに関するポリシーチェックが行われません。	－	0以上

（20）CanonicalizationMethod

正規化アルゴリズムに関する情報を指定します。コンテンツはありません。

表 C-70　CanonicalizationMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	受信側で処理可能な正規化アルゴリズムの URI を指定します。	anyURI	1

（21）SignatureTarget

署名個所に関するポリシーチェックの情報を指定します。

表 C-71　SignatureTarget

種別	要素	説明	データ型	指定回数
属性	Part	署名個所となる SOAP エンベロープ中のエレメントを指定します。メッセージの署名個所がこの属性に指定した個所と異なる場合，ポリシー違反として SOAP Fault をスローします。指定できる値は "Body" だけです。	enum	1
コンテンツ	TransformMethodList	トランスフォームアルゴリズムに関する情報を指定します。省略時はトランスフォームアルゴリズムに関するポリシーチェックが行われません。	―	0 または 1

（22）TransformMethodList

トランスフォームアルゴリズムに関する情報を指定します。受信メッセージ中のトランスフォームアルゴリズムが，この要素の TransformMethod タグで指定した，どのアルゴリズムとも一致しない場合は，ポリシーチェック違反として SOAP Fault をスローします。

表 C-72　TransformMethodList

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	TransformMethod	トランスフォームアルゴリズムのアルゴリズム識別子を指定します。省略時はトランスフォームアルゴリズムに関するポリシーチェックが行われません。	anyURI	0以上

（23）TransformMethod

トランスフォームアルゴリズムのアルゴリズム識別子を指定します。コンテンツはありません。

表 C-73　TransformMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	受信側で処理可能なトランスフォームアルゴリズムの URI を指定します。	anyURI	1

（24）DecryptionPolicyConfig

復号化に関するポリシーチェックの情報を指定します。DecryptionConfig 要素の指定があり，暗号化がされていないメッセージを受信した場合は，ポリシーチェック違反として SOAP Fault をスローします。

表 C-74　DecryptionPolicyConfig

種別	要素	説明	データ型	指定回数
属性	―	―	―	―
コンテンツ	DecryptionTarget	復号化個所に関するポリシーチェックの情報を指定します。	―	1

（25）DecryptionTarget

復号化個所に関するポリシーチェックの情報を指定します。

表 C-75　DecryptionTarget

種別	要素	説明	データ型	指定回数
属性	Part	復号化する個所となる SOAP エンベロープ中のエレメントを指定します。メッセージの復号化個所がこの属性に指定した個所と異なる場合ポリシー違反として SOAP Fault をスローします。指定できる値は "Body" だけです。	enum	1
	Type	復号化する個所の暗号化タイプを指定します。指定可能な値は "Content" だけです。	enum	1
コンテンツ	DecryptionConfig	復号化に関するポリシーチェックの情報を指定します。	―	1

（26）DecryptionConfig

復号化に関するポリシーチェックの情報を指定します。DecryptionConfig 要素の指定があり，暗号化がされていないメッセージを受信した場合は，ポリシーチェック違反として SOAP Fault をスローします。

表 C-76　DecryptionConfig

種別	要素	説明	データ型	指定回数
属性	—	—	—	—
コンテンツ	ContentsDecryption	メッセージ内容の復号化に関するポリシーチェックの情報を指定します。	—	1
	KeyDecryption	復号に使用する鍵の復号化に関するポリシーチェックの情報を指定します。EncryptionType が "ContentsEncryption" の場合は不要です。	—	1

（27）ContentsDecryption

メッセージ内容の復号化に関するポリシーチェックの情報を指定します。

表 C-77　ContentsDecryption

種別	要素	説明	データ型	指定回数
属性	—	—	—	—
コンテンツ	DecryptionMethodList	復号化個所の暗号アルゴリズムに関するポリシーチェックの情報を指定します。省略時は復号化個所の暗号アルゴリズムに関するポリシーチェックが行われません。	—	0 または 1

（28）DecryptionMethodList

復号化個所の暗号アルゴリズムに関するポリシーチェックの情報を指定します。受信メッセージ中の暗号アルゴリズムが，この要素の DecryptionMethod タグで指定した，どのアルゴリズムとも一致しない場合は，ポリシーチェック違反として SOAP Fault をスローします。

表 C-78　DecryptionMethodList

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	DecryptionMethod	復号化に用いる暗号アルゴリズムに関する情報を指定します。省略時は復号化に用いる暗号アルゴリズムに関するポリシーチェックが行われません。	－	0以上

（29）DecryptionMethod

復号化に用いる暗号アルゴリズムに関する情報を指定します。コンテンツはありません。

表 C-79　DecryptionMethod

種別	要素	説明	データ型	指定回数
属性	Algorithm	受信側で処理可能な暗号アルゴリズムの URI を指定します。	anyURI	1

（30）KeyDecryption

復号に使用する鍵の復号化に関するポリシーチェックの情報を指定します。

表 C-80　KeyDecryption

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	DecryptionMethodList	復号化個所に関するポリシーチェックの情報を指定します。省略時は復号化個所に関するポリシーチェックが行われません。	－	0 または 1

（31）TimestampConfig

タイムスタンプに関するポリシーチェックの情報を指定します。

表 C-81　TimestampConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－

種別	要素	説明	データ型	指定回数
コンテンツ	Created	タイムスタンプ要素の Created 要素に関するポリシーチェックの情報を指定します。省略時は Created 要素に関するポリシーチェックが行われません。	―	0 または 1
	Expires	タイムスタンプ要素の Expires 要素に関するポリシーチェックの情報を指定します。省略時は Expires 要素に関するポリシーチェックが行われません。	―	0 または 1

（32）Created

タイムスタンプ要素の Created 要素に関するポリシーチェックの情報を指定します。コンテンツはありません。

表 C-82　Created

種別	要素	説明	データ型	指定回数
属性	Required	タイムスタンプ要素内の Created 要素の有無をチェックするかどうかを "true"，"false"，"1 ”，"0" のどれかで指定します。"true"，"1" を指定した場合タイムスタンプ要素内の Created 要素の有無をチェックし，存在しない場合はポリシー違反として SOAP Fault をスローします。"false"，"0" を指定した場合タイムスタンプ要素内の Created 要素の有無をチェックしません。	boolean	1

（33）Expires

タイムスタンプ要素の Expires 要素に関するポリシーチェックの情報を指定します。コンテンツはありません。

表 C-83　Expires

種別	要素	説明	データ型	指定回数
属性	Required	タイムスタンプ要素内の Expires 要素の有無をチェックするかどうかを "true"，"false"，"1 ”，"0" のどれかで指定します。"true"，"1" を指定した場合タイムスタンプ要素内の Expires 要素の有無をチェックし，存在しない場合はポリシー違反として SOAP Fault をスローします。"false"，"0" を指定した場合タイムスタンプ要素内の Expires 要素の有無をチェックしません。	boolean	1

（34）ResponseReceiverConfig

レスポンスメッセージ受信時の設定を指定します。

表 C-84　ResponseReceiverConfig

種別	要素	説明	データ型	指定回数
属性	－	－	－	－
コンテンツ	ReceiverPortConfig	Web サービスセキュリティポリシー定義を適用する SOAP サービスエンドポイントの情報を指定します。	－	1以上

付録 D　用語解説

このマニュアルで使用している用語の意味を説明します。

（英数字）

JAAS

J2SE が提供する標準的なユーザー認証用の API です。JAAS の API によるユーザー認証を JAAS 認証といいます。

SOAP

分散ネットワーク環境の中で XML ベースの情報を交換するために使用する通信プロトコルの名称です。

SOAP アプリケーション

SOAP および WSDL の技術を利用して開発し，ネットワークを利用して公開，実行できるアプリケーションのことをいいます。SOAP アプリケーションでは，クライアント側に実装された処理によって，SOAP サービスを呼び出し，提供されるサービスを利用します。

SOAP アプリケーション開発支援機能

Cosminexus が提供する，SOAP アプリケーションを開発するための機能です。SOAP アプリケーション開発支援機能では，ウィザードを使用しながら SOAP アプリケーションを開発できます。

SOAP サービス

SOAP アプリケーションを形成するプログラムのうち，サーバ側に配置して，クライアントから要求された処理を実行するプログラム（サービス）のことをいいます。

SOAP 通信基盤

Cosminexus が提供する，SOAP アプリケーションを実行し，SOAP による通信を実現するための環境です。

SOAP ヘッダ

SOAP メッセージの要素です。SOAP ヘッダでは，メッセージ処理のあて先，およびメッセージ処理が必要かどうかを指定します。

SOAP ボディ

SOAP メッセージの要素です。SOAP ボディに送信するメッセージの内容を記述します。

SOAP メッセージ

SOAP プロトコルでオブジェクト間の送受信に使用するメッセージです。SOAP メッセージは，SOAP エンベロープ，SOAP ヘッダ，および SOAP ボディという要素で構成されます。

Web サービス

Web 関連の技術を利用して，ネットワークを介して提供されるサービスです。Web サービスの基礎

となる技術には，SOAP，WSDL，および UDDI などがあります。

Web サービスセキュリティ

このマニュアルでは，WS-Security 仕様に基づいた，SOAP メッセージに対するセキュリティ技術を Web サービスセキュリティと呼びます。

Web サービスセキュリティ機能

Cosminexus が提供する，Web サービスセキュリティを実現するための機能です。Web サービスセキュリティ機能は，Cosminexus Web Services - Security が提供する定義ファイルを設定することによって，使用できます。

Web サービスセキュリティ機能定義ファイル

Cosminexus が提供する Web サービスセキュリティ機能の詳細を定義するための XML ファイルです。Web サービスセキュリティ機能定義ファイルは，サーバとクライアントの両方に配置します。

Web サービスセキュリティポリシー定義ファイル

Web サービスセキュリティ機能を使用する場合に従うポリシーを定義するための XML ファイルです。Web サービスセキュリティポリシー定義ファイルは，サーバとクライアントの両方に配置します。

WS-Security

OASIS が規定している，Web サービスのセキュリティに関する仕様書です。WS-Security は，XML セキュリティの技術を利用しています。

XML セキュリティ

このマニュアルでは，W3C が規定している XML 署名および XML 暗号を総称して XML セキュリティと呼びます。

（カ行）

環境設定ファイル

サーバやクライアントの実行環境に合わせて，必要な設定を変更するためのプロパティファイルです。環境設定ファイルには，キーストアファイルや証明書ファイルの格納場所などを記述します。

共通鍵

共通鍵暗号で使用する，決められた 2 者間だけで共有する鍵（secret key）です。共通鍵暗号では，共通鍵でデータを復号化します。共通鍵は，決められた 2 者が安全に管理します。

公開鍵

公開鍵暗号で使用する，データの暗号化または電子署名を検証するのための鍵です。公開鍵は，通常ネットワーク上などで公開されています。データを暗号化して送信する場合，送信者は受信者の公開鍵を使用してデータを暗号化します。また，受信した電子署名を検証する場合は，受信者は送信者の公開鍵を使用して電子署名を検証します。

コマンドライン Java アプリケーション

コマンドラインから起動して使用する Java アプリケーションです。

（サ行）

証明書

オープンな企業情報システム上で情報をやり取りするときに，通信相手が本人であることを証明するための電子的な情報です。暗号技術を用いることで，他人が成り済ますことができないようになっています。Web サービスセキュリティ機能では，X.509 証明書を扱えます。

セキュリティトークン要素

WS-Security で規定されている，セキュリティトークンに関する要素です。Web サービスセキュリティ機能で扱うセキュリティトークン要素は，UsernameToken 要素と BinarySecurityToken 要素です。

セキュリティヘッダ

SOAP ヘッダに含まれる wsse:Security という名称の要素です。セキュリティヘッダはセキュリティ要素で構成されます。

セキュリティ要素

SOAP メッセージに含まれる，セキュリティに関する要素です。このマニュアルでは，セキュリティトークン要素，署名要素，暗号化要素などを総称してセキュリティ要素と呼びます。

（タ行）

デプロイ定義ファイル

Web サービスセキュリティ機能を利用するために必要な情報を読み込む処理を定義する XML ファイルです。デプロイ定義ファイルには，サーバ用のファイルとクライアント用のファイルの 2 種類があります。

（ハ行）

秘密鍵

公開鍵暗号で使用する，データの復号化または電子署名を作成するのための鍵（private key）です。非公開鍵ともいいます。秘密鍵は，その所有者が安全に管理します。
暗号化されたデータを受信した場合，受信者は自分の秘密鍵を使用してデータを復号化します。また，送信するデータに電子署名を付加する場合は，送信者は自分の秘密鍵を使用し電子署名を作成します。

索引

H

J

K

L

M

N

P

R

S

T

U

V

W

X

あ

い

か

き

く

こ

さ

し

せ

て

と

に

ひ

ふ

め

ろ

ソフトウェアマニュアルのサービス ご案内

１．マニュアル情報ホームページ

ソフトウェアマニュアルの情報をインターネットで公開しています。

URL　http://www.hitachi.co.jp/soft/manual/

ホームページのメニューは次のとおりです。

■マニュアル一覧	日立コンピュータ製品マニュアルを製品カテゴリ，マニュアル名称，資料番号のいずれかから検索できます。
■CD-ROMマニュアル	日立ソフトウェアマニュアルと製品群別CD-ROMマニュアルの仕様について記載しています。
■マニュアルのご購入	マニュアルご購入時のお申し込み方法を記載しています。
■オンラインマニュアル	一部製品のマニュアルをインターネットで公開しています。
■サポートサービス	ソフトウェアサポートサービスお客様向けページでのマニュアル公開サービスを記載しています。
■ご意見・お問い合わせ	マニュアルに関するご意見，ご要望をお寄せください。

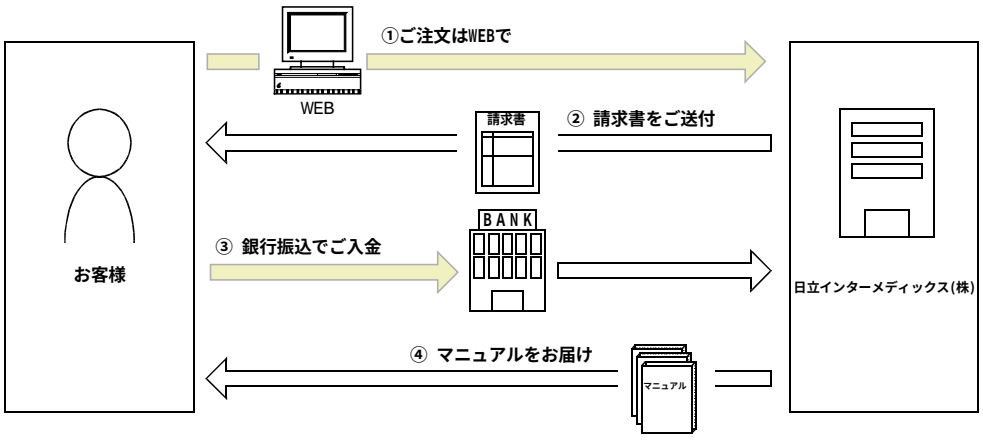
２．インターネットでのマニュアル公開

2 種類のマニュアル公開サービスを実施しています。

(1) マニュアル情報ホームページ「オンラインマニュアル」での公開
　製品をよりご理解いただくためのご参考として，一部製品のマニュアルを公開しています。

(2) ソフトウェアサポートサービスお客様向けページでのマニュアル公開
　ソフトウェアサポートサービスご契約のお客様向けにマニュアルを公開しています。公開しているマニュアルの一覧，本サービスの対象となる契約の種別などはマニュアル情報ホームページの「サポートサービス」をご参照ください。

３．マニュアルのご注文



① マニュアル情報ホームページの「マニュアルのご購入」にアクセスし，お申し込み方法をご確認のうえ WEB からご注文ください。ご注文先は日立インターメディックス(株)となります。

② ご注文いただいたマニュアルについて請求書をお送りします。

③ 請求書の金額を指定銀行へ振り込んでください。

④ 入金確認後 7 日以内にお届けします。在庫切れの場合は，納期を別途ご案内いたします。