

Metasign-server

Définition des interfaces



Version :	1.23
Date de document:	10 octobre 2016
Ref. Doc.:	MSIGN-SRV-GDE-02

Statut

Rédaction	VKA
Validation	VKA
Classification	Publique
Etat du document	Validé
Version actuelle	1.23
Référence	MSIGN-SRV-GDE-02
Version Produit applicable	2.0.0

Diffusion

Diffusion publique

Historique des révisions

Date	Version	Commentaires
04/10/2012	0.1	Création du document
18/10/2012	1.0	Version valide. Applicable à la version 1.0.0 du serveur de signature.
14/01/2013	1.1	Version applicable à partir de la version 1.1.0 du serveur de signature Ajout des descriptifs des opérations du Web Service Admin
06/02/2013	1.2	Ajout des descriptifs pour les opérations : - Signature d'un document - Augmentation d'une signature

03/04/2013	1.3	<p>Modification des opérations suivantes pour ajouter l'identifiant de clés de signature :</p> <ul style="list-style-type: none"> - createSignatureKeyFromPKCS12 - generateSignatureKeyAndCSR <p>Modification de type de paramètre pour le paramètre signatureKeyId dans l'opération</p> <ul style="list-style-type: none"> - signDocument - generateCSRForSignatureKey <p>Ajout de l'opération suivante:</p> <ul style="list-style-type: none"> - createSignatureKeyCertFromKeyPair
03/06/2013	1.4	<ul style="list-style-type: none"> - Mise à jour des opérations existantes - Ajout de chapitre pour les futures opérations définies.
11/07/13	1.5	<ul style="list-style-type: none"> - Mise à jour des nouvelles opérations
07/10/13	1.6	<ul style="list-style-type: none"> - Modification des éléments WSOOperation et WSResponse
19/12/13	1.7	<ul style="list-style-type: none"> - Mise à jour en accord avec la version 1.1.2 du serveur de signature (opération de récupération des politiques de signature, liste des profils de signature, liste des profils de génération de clés de signature) - Correction des descriptions relatives aux types « WSOOperation » et « WSResponse » pour décrire l'héritage mis en œuvre dans les WSDL - Ajout d'exemple sur les opérations de mise à jour des utilisateurs et la consultation des groupes et des utilisateurs - Ajout d'un chapitre décrivant l'utilisation des critères de recherche pour les opérations de consultation.
02/02/15	1.8	<p>Mise à jour en accord avec la version 1.3.0 du serveur de signature :</p> <ul style="list-style-type: none"> - Retrait de l'opérateur booléen XOR des interfaces de recherche. (sections 5.17 et 5.18). - Ajout de l'opération ListSignerId au WS SigOps §2.6 - Ajout des informations optionnelles pour la signature visuelle §5.7 modifié
05/03/15	1.9	<p>Ajout de l'option de génération du rapport de vérification pour les signatures.</p>
12/03/15	1.10	<p>Précision du non support du format PADES-CMS (§21).</p>
21/04/15	1.11	<p>Pour permettre la prise en compte du traitement des clés avec OTP.: Ajout de l'opération ActivateSignatureKey (§58), de BullC2PActivationSecret (§Erreur : source de la référence non trouvée) et de SkGenerationProfileSpecification (§105). Ajout §119, 121 et 122 .</p>
07/05/15	1.12	<p>Ajout de l'attribut 'maxResults' pour limiter le nombre de réponses renvoyées par les requêtes 'ListXXX'. Modification (§116).</p>
09/06/15	1.13	<p>Modification du §4.9 pour indiquer qu'à partir de la version 1.3.0, les certificats d'authentification des utilisateurs seront renvoyés par l'opération ListUsers de WS Admin.</p>

11/06/15	1.14	Retrait des modifications sur les certificats X509 au sein des réponses du ListUsers de WS Admin. Ajout du paramètre de retour « totalResults » dans les réponses à ListUsers et ListGroups. Ajout de l'opération ConsultUser dans WS Admin. Ajout du type « ServerId2NameInfo ».
17/06/15	1.15	Rajout de l'élément « trustedApplications » dans le type « ApplicationSpecification » §5.13.2
16/10/15	1.16	Ajout de l'attribut « UserTypeFilter » dans l'opération ListUsers §79
20/10/15	1.17	Mise à jour au format Atos
04/01/16	1.18	Ajout de l'attribut « CredentialKey » dans l'élément CredentialDescription. Correction de la description de l'attribut ApplicationReference §5.13.1. L'opération activateSignatureKey a été déplacé vers le WS de Gestion de profile et politique de signature.
22/01/16 22/02/16 17/03/16	1.19	Ajouts de nouveaux éléments dans l'élément « DASAAuthentication » Ajout de l'opération « requestCertificateForSignatureKey » dans le WS de Gestion de profile et politique de signature §3.24. Mise à jour des namespaces.
31/05/16 06/10/16	1.20	Ajout des paramètres de configuration du visuel de signature Support des nouveaux formats EIDAS.
13/02/17 14/03/17	1.21	Ajout du support des usages de clés associés aux clés de signature. Ajout des certificats de signature en résultat de ConsultUser.
29/05/18	1.22	Mise à jour de l'opération activateSignatureKey
08/06/18	1.23	Ajout des interfaces pour la gestion des token FIDO

Sommaire

1	Introduction.....	8
1.1	Présentation du contexte.....	8
1.2	Références documentaires	9
1.3	Glossaire.....	10
2	Service Web : Gestion des opérations de signature	11
2.1	Dépôt d'un document ou d'une signature sur le serveur	11
2.2	Récupération d'un document ou d'une signature sur le serveur.....	13
2.3	Signature d'un document.....	14
2.4	Augmentation d'une signature	17
2.5	Vérification d'une signature	20
2.6	Récupération de l'identifiant d'un signataire	24
3	Service Web : Gestion des profils, des politiques de signature	26
3.1	Dépôt de politique de signature	26
3.2	Récupération d'une politique de signature.....	28
3.3	Suppression d'une politique de signature.....	29
3.4	Récupération d'une liste de politiques de signature présents sur le serveur	30
3.5	Dépôt d'un profil de signature	32
3.6	Mise à jour d'un profil de signature	34
3.7	Récupération d'un profil de signature	35
3.8	Suppression d'un profil de signature	37
3.9	Récupération d'une liste de profils de signature présents sur le serveur	38
3.10	Dépôt d'un profil de génération de clé de signature	39
3.11	Mise à jour d'un profil de génération de clé de signature	41
3.12	Récupération d'un profil de génération de clé de signature	42
3.13	Suppression d'un profil de génération de clé de signature	43
3.14	Récupération d'une liste de profils de génération de clés de signature présents sur le serveur	44
3.15	Mise à jour de secret d'activation de clé de signature d'un signataire	46
3.16	Dépôt d'un container de clé PKCS#12.....	47
3.17	Création d'une clé de signature à partir d'un PKCS#12.....	48
3.18	Création d'une clé de signature à partir d'un identifiant de clé dans le HSM.....	50
3.19	Création d'une clé de signature à partir d'un profil de clé de signature.....	52
3.20	Suppression d'une clé de signature	53
3.21	Création d'une requête de certificat pour une clé de signature existante	55

3.22	Dépôt d'un certificat de signature pour une clé de signature existante	57
3.23	Activation d'une clé de signature.....	58
3.24	Génération de certificat de signature pour une clé de signature existante.....	60
4	Service Web : Administration des utilisateurs du serveur de signature	62
4.1	Création d'un utilisateur	62
4.2	Mise à jour d'un utilisateur	64
4.3	Supprimer un utilisateur	69
4.4	Créer un groupe	70
4.5	Supprimer un groupe	72
4.6	Ajouter des utilisateurs dans un groupe	73
4.7	Retirer des utilisateurs dans un groupe	75
4.8	Consulter une liste de groupes existant.....	76
4.9	Consulter une liste d'utilisateurs existants	79
4.10	Consulter un utilisateur.....	85
4.11	Initialisation du processus d' enrôlement FIDO	89
4.12	Dépôt d'un token FIDO pour un utilisateur.....	91
4.13	Récupération des identifiants des token FIDO.....	92
4.14	Suppression d'un token FIDO	93
5	Définition des éléments complexes des paramètres des Web services	95
5.1	Type « WSOperation »	95
5.2	Type « WSResponse »	95
5.3	Type « Document »	96
5.4	Type « SignerCertificate »	97
5.5	Type « SignVerifReport »	97
5.6	Type « SignatureProfile »	98
5.7	Type « SignatureOptionalInfos »	101
5.8	Type « SkGenerationProfile ».....	105
5.9	Type « ArtifactInfoFile »	107
5.10	Type « Keystore »	108
5.11	Type « ActivationSecret »	108
5.12	Type « User »	110
5.13	Type « Application ».....	111
5.14	Type « Credential »	113
5.15	Type « Group ».....	114
5.16	Type « UsersToGroupAssociation »	115
5.17	Type « Chain »	116

5.18	Type EnforcedAuthenticationMethod	119
5.19	Type ActivationData	121
5.20	Type ActivationResult	122
5.21	Type ServerId2NameInfo	123
5.22	Type KeyUsageArray	123
5.23	Type Simples	125
5.24	Type CertificateRequestParameters	125
5.25	Type RequestCertificateResult	126
5.26	Type SCEPRequestResult	126
5.27	Type « FIDOInfos »	127
5.28	Type « RegisteredToken »	128
6	Critères de recherche possibles pour les différents objets	129
6.1	Recherches d'utilisateurs (ou d'applications)	129
6.2	Recherches de groupes	129
6.3	Recherches de politiques de signature	129
6.4	Recherches de profils de signature	130
6.5	Recherches de profils de génération de clé de signature	130

1 Introduction

1.1 Présentation du contexte

Ce document présente les interfaces externes d'utilisation du serveur de signature électronique MetaSIGN-SERVER. Ce serveur fournit des services de signature électronique et de vérification de signatures électroniques disponibles sous forme de web services.

Les services du serveur de signature sont invocables par une application cliente sous forme de services web, en utilisant le protocole SOAP 1.1 sur HTTPS.

Ce document décrit les opérations exposées par MetaSIGN-SERVER et les types des arguments utilisés en entrée et en retour pour ces opérations.

Ce document de spécification des interfaces d'appels aux services web doit permettre à des programmeurs en charge de développer une application qui réalise des opérations de signatures numériques avancées ou de vérifications de signatures pour le compte d'un utilisateur ou d'une entreprise. MetaSIGN-SERVER offre aussi les services web permettant d'administrer les utilisateurs du serveur de signature et des services associés.

Les chapitres suivant décrivent chacune des opérations pour les services Web du serveur suivants :

- Gestion des opérations de signature :
 - Dépôt d'un document ou d'une signature sur le serveur;
 - Récupération d'un document ou d'une signature sur le serveur;
 - Signature d'un ou plusieurs documents;
 - Augmentation d'une signature ;
 - Vérification d'une signature.
- Gestion des profils et des politiques de signature :
 - Dépôt d'une politique de signature;
 - Dépôt d'un profil de signature;
 - Dépôt d'un profil de génération de clé de signature;
 - Mise à jour d'un profil de signature existant.
- Administration des utilisateurs du serveur de signature :
 - Création d'un utilisateur;
 - Mise à jour d'un utilisateur ;
 - Récupération des token FIDO d'un utilisateur;
 - Suppression d'un token FIDO ;
 - Dépôt d'un token FIDO d'un utilisateur ;
 - Enrôlement d'un utilisateur FIDO (signataire).

Les documents standards WSDL (Web Service Description Language) décrivant les services web offerts par MetaSIGN-SERVER permettant à un client SOAP de se configurer de manière automatique (en « parlant » ce document) se trouvent sous les références suivantes :

- Gestion des opérations de signature : SigOps.wsdl
- Gestion des profils et des politiques de signature : AdminSig.wsdl
- Administration des utilisateurs du serveur de signature : Admin.wsdl

1.2 Références documentaires

1.2.1 Références internes

Référence	Titre	Version
MSIGN-PS-02	Description des politiques de signature au format XML	1.5
MSIGN-API-GDE-03	Description Rapport de vérification de la signature	2.0
SigOps.wsdl	Document WSDL décrivant les services Web de gestion des opération de signature	
AdminSig.wsdl	Document WSDL décrivant les services Web de gestion des profils et des politiques de signature	
Admin.wsdl	Document WSDL décrivant les services Web d'administration des utilisateurs et du serveur de signature	

1.2.2 Références externes

Référence	Titre	Document	Version
TS 102 778-1	PDF Advanced Electronic Signature Profiles	Part 1: PAdES Overview - a framework document for PAdES	1.1.1
TS 102 778-2	PDF Advanced Electronic Signature Profiles	Part 2: PAdES Basic - Profile based on ISO 32000-1	1.2.1
TS 102 778-3	PDF Advanced Electronic Signature Profiles	Part 3: PAdES Enhanced – PAdES-BES and PAdES-EPES Profiles	1.2.1
TS 102 778-4	PDF Advanced Electronic Signature Profiles	Part 4: PAdES Long Term - PAdES-LTV Profile	1.1.2

ISO 32000-1	Portable document format	Portable document format — Part 1: PDF 1.7	2008
TS 101 733	CMS Advanced Electronic Signatures (CAAdES)	Electronic Signatures and Infrastructures (ESI)	1.6.3
draft-nourse-scep	Simple Certificate Enrollment Protocol (SCEP)	https://tools.ietf.org/html/draft-nourse-scep-23	23

1.3 Glossaire

Acronymes	Définition
WS	Web Service
WSDL	Web Service Description Language
XSD	XML Schema Definition
XML	Extensible Markup Language (langage de balisage extensible)
URL	Uniform Resource Locator (localisateur uniforme de ressource)
SCEP	Simple Certificate Enrollment Protocol (protocole simple d'enregistrement de certificat)

2 Service Web : Gestion des opérations de signature

Ce chapitre décrit toutes les opérations exposées par le service Web de gestion des opérations de signature.

Elles permettent à un signataire ou une application de signature de réaliser les opérations de signature sur le serveur de signatures.

- URL du Service :

<https://hostname:port/servlets/SignServer/com.bull.security.signserver.ws.sigops/services/sigOpsSOAP>

« Hostname » est le nom du serveur (ou adresse IP) sur lequel le serveur de signature MetaSIGN-Serveur est déployé

« Port » est le numéro de port écouté par le serveur.

- Espaces de nommages :

```
xmlns:tns="http://www.bull.security.com/SignServer/sigOps/service/v1.3.0/"
```

```
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
```

```
xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
```

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

```
xmlns:sigops="http://www.bull.security.com/SignServer/sigOps/v1.3.0/">
```

2.1 Dépôt d'un document ou d'une signature sur le serveur

2.1.1 Description

Cette opération permet à l'utilisateur d'envoyer un document ou une signature au serveur de signature pour un usage futur dans une ou plusieurs opération de signature ou de vérification.

Le document (ou la signature) est envoyé dans la requête, le serveur de signature le sauvegarde et crée un identifiant qui lui est attaché. Cet identifiant unique de document est ensuite rendu en retour de la requête à l'utilisateur pour son usage futur dans une autre opération.

Il est possible de définir (optionnellement) un utilisateur (par son identifiant) qui pourra utiliser l'identifiant du document déposée pour réaliser une autre opération (de signature ou de vérification) sur celui-ci. Aucun autre utilisateur (à l'exception de celui identifié et de l'application ayant déposé le document) ne pourra alors effectuer des opérations sur ce document.

Note : Dans la version 1.1.2 de MetaSIGN-SERVER, le mécanisme de gestion des droits associés n'est pas opérationnel. De fait, tout utilisateur authentifié et connaissant l'identifiant du document déposé, pourra effectuer une opération sur celui-ci.

Nom de l'opération exposé : **depositDocumentOrSignature**

Identifiant SOAP : **depDocSig**

2.1.2 Paramètres d'entrée

Elément de type WSDepDocSig (extension de WSOperation)

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut Optionnel - Type : ServerID 	voir descriptif du type WSOperation au paragraphe 5.1.
bytes	<ul style="list-style-type: none"> - Elément - Obligatoire - base64Binary 	Contient le document à déposer encodé en Base64.

2.1.3 Paramètres de sortie

Elément de type DepDocSigResponse (extension de WSResponse)

Identifiant SOAP	Propriété	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut Obligatoire - Type : ReturnStatusInfo 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.

docId	<ul style="list-style-type: none"> - Attribut Optionnel - Type : ServerId 	indique l'identifiant qui a été associé au document déposé.
-------	---	---

2.2 Récupération d'un document ou d'une signature sur le serveur

Disponible à partir de la version 1.1.0

2.2.1 Description

Cette opération permet à l'utilisateur de récupérer un document ou une signature se trouvant sur le serveur de signature. Celui-ci a soit été préalablement déposé par l'application ou l'utilisateur, soit construit lors d'une opération sur le serveur de signature. Cette opération est notamment utilisée après une opération de signature ou d'augmentation de signature.

L'identifiant du document (ou la signature) doit être envoyé dans la requête, le serveur de signature recherche le document ou la signature associée à cet identifiant. Celui-ci est ensuite envoyé en retour de la requête à l'utilisateur.

Nom de l'opération exposé : **retrieveDocumentOrSignature**

Identifiant SOAP : **retDocSig**

2.2.2 Paramètres d'entrée

Elément retDocSig de type WSRetDocSig (extension de WSOperation)

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
id	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ServerId 	Indique l'identifiant associé au document ou à la signature à récupérer. (voir descriptif au paragraphe 125)

2.2.3 Paramètres de sortie

Elément de type DocContentResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
docContent	- Elément - Optionnel - Type : DocContent	Contenu du document. Il contient un élément bytes où se trouve le contenu encodé en « Base64 » du document.

2.3 Signature d'un document

Disponible à partir de la version 1.1.0 en mode non différé seulement

2.3.1 Description

Cette opération permet à l'utilisateur de signer un document.

Pour toutes les opérations de signature, sauf restriction précisée dans la description de l'opération :

- les trois formats de signature sont accessibles (CADES, XAdES ou PAdES) ,

- les deux modes de réponse : synchrone ou asynchrone peuvent être utilisés

De plus, lors d'une opération de signature et si le profil de signature le requiert, une augmentation de signature sera effectuée de façon transparente et la signature augmentée sera retournée (ou créée sur le serveur en mode asynchrone).

Le document à signer et/ou la signature peut être transmis selon plusieurs modes :

- Le document à signer est émis dans la requête;
- Le document a été précédemment téléversé sur le serveur. son identifiant est alors transmis;
- Seul le haché du document (calculé par l'appelant et sous sa responsabilité) peut être transmis; (uniquement pour une signature détachée sans augmentation);
- Le document à signer se trouve sur une URL.

Lors d'une opération de signature le secret d'activation de la clé de signature est demandé.

Nom de l'opération exposé : **signDocument**

Identifiant SOAP : **sigDoc**

2.3.2 Paramètres d'entrée

Élément sigDoc de type WSSignDocument (extension de WSSignatureOperation et WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
deferredMode	- Attribut Optionnel - Type : boolean	Paramètre optionnel contenant le mode de retour de la signature du document. Si la valeur est True alors la réponse de la requête contiendra l'identifiant de la signature. Si la valeur est False alors la réponse de la requête contiendra la signature.

Identifiant SOAP	Propriété	Description
includeVerificationReport	<ul style="list-style-type: none"> - Attribut Optionnel - Type : boolean 	<p>Paramètre optionnel permettant lorsqu'il est à « true » de forcer la génération d'un rapport de vérification de la signature générée. Ce rapport est alors renvoyé dans la réponse du web service.</p> <p>NB :</p> <ul style="list-style-type: none"> • lorsque l'attribut est null ou à « false », aucun rapport de vérification ne sera généré (ce qui permet de réaliser l'opération plus rapidement) • lorsque que la signature demandée n'est pas augmentée, l'attribut est silencieusement ignoré : aucun rapport de vérification ne sera renvoyé.
, signatureSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ActivationSecret 	<p>Élément complexe contenant le secret d'activation de la clé de signature qui sera utilisée pour signer le document (voir descriptif de l'élément au paragraphe 5.11).</p>
signatureKeyId	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : SignatureKeyId (string) 	<p>Paramètre contenant l'identifiant de la clé de signature à utiliser pour signer le document.</p> <p>(voir la description au paragraphe 125)</p>
signatureProfile	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : SignatureProfile 	<p>Élément complexe contenant les informations sur le profil de signature qui sera utilisé pour signer le document(voir descriptif de l'élément au paragraphe 5.6).</p>
optionalInfos	<ul style="list-style-type: none"> - Élément - Optionnel - Type : SignatureOptionalInfos 	<p>Élément complexe contenant des informations optionnelles pouvant être ajouté à la signature (voir descriptif de l'élément au paragraphe 5.7).</p>
document	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Document 	<p>Élément complexe permettant de donner le document à signer (voir descriptif de l'élément au paragraphe 5.3).</p>

2.3.3 Paramètres de sortie

Élément sigDocResponse de type SignDocResponse (extension de WSResponse).

Identifiant SOAP		Type	Description
returnStatusEnum		- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo		- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
En fonction de la valeur de « deferredMode »	signatureReport	- Élément - Optionnel - Type : SignatureReport	Élément complexe contenant le rapport de la signature (voir descriptif de l'élément au paragraphe 5.5). Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « False ».
	reportId	- Attribut - Obligatoire - Type : ServerID (string)	Paramètre contenant l'identifiant durapport de la signature. Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « True ».
deferredMode »	SignatureId	- Attribut - Obligatoire - Type : ServerID (string)	Paramètre contenant l'identifiant de la signature générée. Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « True ».

Identifiant SOAP		Type	Description
	SignatureContent	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : docContent 	Contient la signature encodé en « Base64 ». Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « False ».

2.4 Augmentation d'une signature

Disponible à partir de la version 1.1.0 en mode non différé seulement

2.4.1 Description

Cette opération permet à l'utilisateur d'augmenter une signature d'un document.

Pour toutes les opérations d'augmentation de signature, sauf restriction précisée dans la description de l'opération :

- les trois formats de signature sont accessibles (CAdES, XAdES ou PAdES) ,
- les deux modes de réponse : synchrone ou asynchrone peuvent être utilisés

Lors d'une opération d'augmentation de signature, le profil de signature doit contenir les informations sur les éléments d'augmentation.

Le document et la signature à augmenter peuvent être transmis selon plusieurs modes :

- Le document et/ou la signature est émis dans la requête;
- Le document et/ou la signature a été précédemment téléversé sur le serveur. Son identifiant est alors transmis;
- Seul le haché du document (calculé par l'appelant et sous sa responsabilité) peut être transmis; (uniquement pour une signature détachée);
- Le document et/ou la signature se trouve sur une URL.

Lors d'une opération d'augmentation de signature le secret d'activation de la clé de signature est demandé.

Nom de l'opération exposé : **augmentSignature**

Identifiant SOAP : **augSig**

2.4.2 Paramètres d'entrée

Elément augSig de type WSAugmentSignature (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
deferredMode	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	<p>Paramètre optionnel contenant le mode de retour de la signature du document.</p> <p>Si la valeur est True alors la réponse de la requête contiendra l'identifiant de la signature.</p> <p>Si la valeur est False alors la réponse de la requête contiendra la signature.</p>
signature	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Document 	Elément complexe permettant de donner la signature à augmenter (voir descriptif de l'élément au paragraphe 5.3).
document	<ul style="list-style-type: none"> - Élément - Optionnel - Type : Document 	Elément complexe permettant de donner le document dont la signature doit être augmentée. Ce paramètre est nécessaire uniquement dans le cas d'une signature détachée (voir descriptif de l'élément au paragraphe 5.3).
signatureProfile	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : SignatureProfile 	Elément complexe contenant les informations sur le profil de signature qui sera utilisé pour signer le document (voir descriptif de l'élément au paragraphe 5.6).
optionalInfos	<ul style="list-style-type: none"> - Élément - Optionnel - Type : SignatureOptionalInfos 	Elément complexe contenant des informations optionnelles pouvant être ajouté à la signature (voir descriptif de l'élément au paragraphe 5.7).

2.4.3 Paramètres de sortie

Élément auSigResponse de type SignDocResponse (extension de WSResponse).

Identifiant SOAP		Type	Description
returnStatusEnum		- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo		- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
En fonction de la valeur de « deferredMode »	signatureReport	- Élément - Optionnel - Type : signatureReport	Élément complexe contenant le rapport de la signature (voir descriptif de l'élément au paragraphe 5.5). Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « False ».
	reportId	- Attribut - Obligatoire - Type : ServerID (string)	Paramètre contenant l'identifiant du rapport de la signature. Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « True ».
deferredMode »	SignatureId	- Attribut - Obligatoire - Type : ServerID (string)	Paramètre contenant l'identifiant de la signature générée. Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « True ».

Identifiant SOAP		Type	Description
	SignatureContent	- Elément - Obligatoire - Type : docContent	Contient la signature encodé en « Base64 ». Ce paramètre est valorisé si le paramètre d'entrée de l'opération vaut « False ».

2.5 Vérification d'une signature

2.5.1 Description

Cette opération permet à l'utilisateur de vérifier une signature d'un document, celle-ci étant détachée ou non du document original.

L'utilisateur doit fournir dans la requête la signature et optionnellement le document original (lorsque la signature est détachée) ainsi que l'OID d'une politique de signature qui sera alors utilisée lors de l'opération de vérification.

L'appelant fournit pour la signature soit :

- L'identifiant d'une signature préalablement générée par le serveur ou déposée par un utilisateur via l'opération « `depositDocumentOrSignature` »;
- Une URL qui définit l'endroit où la signature est disponible. Cette URL doit être accessible en mode non authentifié (connexion HTTP uniquement). L'accès à cet URL ne doit pas transiter par un proxy.
- Le contenu de la signature (encodée en base64).
- L'appelant fournit pour le document soit :
- L'identifiant du document préalablement déposé par un utilisateur via l'opération « `depositDocumentOrSignature` »;
- Une URL qui définit l'endroit où le document est disponible. Cette URL doit être accessible en mode non authentifié (connexion HTTP uniquement). L'accès à cet URL ne doit pas transiter par un proxy;

- Le contenu de la signature (encodée en base64);
- Le hash du contenu du document. Ce mode d'opération ne peut être appliqué que lors du vérification d'une signature détachée.

L'identifiant de la politique de signature est utilisé pour vérifier la validité de la signature donnée en paramètre d'entrée. Cet identifiant de politique peut être optionnel lorsque la signature embarque l'identifiant d'une politique. La politique associée à cet identifiant doit avoir été déposée au préalable sur le serveur de signature par l'opération « `depositSignaturePolicy` ».

Nom de l'opération exposé : **verifySignature**

Identifiant SOAP : **verifSig**

2.5.2 Paramètres d'entrée

Elément `verifSig` de type `WSVerifSig` (extension de `WSOperation`).

Identifiant SOAP	Propriété	Description
<code>inDelegationOf</code>	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : <code>ServerId</code> 	Voir descriptif du type <code>WSOperation</code> au paragraphe 5.1.
<code>signatureFormat</code>	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : <code>SignatureFormatForVerification</code> (string) 	Format de la signature qui doit être vérifiée. Valeurs possibles : <ul style="list-style-type: none"> - CADES : pour le format CADES - CMS : pour le format CMS - PADES : pour le format PADES - XADES : pour le format XADES
<code>signature</code>	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : <code>Document</code> 	Elément complexe permettant de donner la signature à vérifier (voir descriptif de l'élément au paragraphe 5.3).
<code>document</code>	<ul style="list-style-type: none"> - Élément - Optionnel - Type : <code>Document</code> 	Elément complexe permettant de donner le document lorsque la signature est détachée (voir descriptif de l'élément au paragraphe 5.3).

Identifiant SOAP	Propriété	Description
policyID	- Élément - Optionnel - Type :string	OID de la politique à utiliser pour la vérification de la signature. cet élément est obligatoire pour certains formats de signature (voir tableaux en page 22)
signerCertificate	- Élément - Optionnel - Type : SignerCertificate	Elément complexe permettant de donner le certificat contenant la clé publique correspond à la clé privée ayant signé le certificat. Ce élément ne doit être renseigné uniquement dans le cas de vérification des signatures au format CMS et PAdES-CMS (non supporté actuellement). (voir descriptif de l'élément au paragraphe 5.4).
verificationLevel	- Élément - Optionnel - Type : AugmentationLevel Type	Elément indiquant le niveau d'augmentation minimal auquel la signature doit être conforme. Les valeurs possibles sont : - B - T - LT - LTA

Présence de référence à l'OID de politique en fonction du format de signature

Format signature	type augmentation		Présence de la référence de l'OID de politique de signature
	Par format	Par niveau	
CMS	pas d'augmentation	-	Obligatoire

Format signature	type augmentation		Présence de la référence de l'OID de politique de signature
	Par format	Par niveau	
CADES-BES	pas d'augmentation	B	Obligatoire
	CADES-T	T	
	CADES-C	LT	
	CADES-XL		
	CADES X1		
	CADES X2		
	CADES A		
CADES-EPES	pas d'augmentation	B	Optionnelle
	CADES-T	T	
	CADES-C	LT	
	CADES-XL		
	CADES X1		
	CADES X2		
	CADES A		
XAdES-BES	pas d'augmentation	B	Obligatoire
	XAdES-T	T	
	XAdES-C	LT	
	XAdES-XL	LTA	
	XADES X1		
	XADES X2		
	XADES A		

Format signature	type augmentation		Présence de la référence de l'OID de politique de signature
	Par format	Par niveau	
XAdES-EPES	pas d'augmentation XAdES-T XAdES-C XAdES-XL XADES X1 XADES X2 XADES A	B T LT LTA	Optionnelle
PAdES-CMS Ce format n'est pas supporté pour le moment	pas d'augmentation	-	Obligatoire
PAdES-BES	pas d'augmentation PAdES-T PAdES-LTV	B T LT LTA	Obligatoire
PAdES-EPES	pas d'augmentation PAdES-T PAdES-LTV	B T LT LTA	Optionnelle

2.5.3 Paramètres de sortie

Elément signVerificationReport de type VerifSigResponse (extension de Wsresponse).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signVerificationReport	- Élément - Optionnel - Type : signatureReport	Élément complexe contenant le rapport de vérification de la signature (voir descriptif de l'élément au paragraphe 5.5).

2.6 Récupération de l'identifiant d'un signataire

2.6.1 Description

Cette opération permet à une application de récupérer l'identifiant d'un signataire dont le nom correspond à celui passé en paramètre de la requête.

L'appelant doit être l'une des applications de confiance du signataire et avoir le rôle « SignManager ».

L'appelant fournit pour la récupération d'identifiant de signataire :

- Le nom du signataire recherché.

Le nom sera celui entré lors de l'enrôlement du signataire une correspondance exacte sera recherchée.

Nom de l'opération exposée : **listSignerId**

Identifiant SOAP : **listSignerId**

2.6.2 Paramètres d'entrée

Élément listSignerId de type WSListSignerId (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOOperation au paragraphe 5.1.
signerName	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : string 	Nom du signataire dont on doit rechercher l'identifiant.

2.6.3 Paramètres de sortie

Élément listSignerIdResponse de type listSignerIdResponse (extension de Wsresponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
signerId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId (string) 	Identifiant du signataire

3 Service Web : Gestion des profils, des politiques de signature

Ce chapitre décrit toutes les opérations exposées par le services Web de gestion des profils, des politiques de signature.

Elles permettent à un signataire ou une application de signature de :

- Définir les politiques de signature (et de vérification) utilisées par le serveur de signature.
- Définir et mettre à jour les profils de signatures qui seront requises pour chaque opération spécifique de signature.
- Définir les profils de génération de clés de signature qui seront requis lorsqu'un utilisateur demande au serveur de signature de générer une clé de signature pour un signataire (ou application de signature).
- URL du Service :

<https://hostname:port/servlets/SignServer/com.bull.security.signserver.ws.adminSig/services/AdminSigSOAP>

« Hostname » est le nom du serveur (ou adresse IP) sur lequel le serveur de signature MetaSIGN-Serveur est déployé

« Port » est le numéro de port écouté par le serveur.

- Espaces de nommages :

```
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:tns="http://www.bull.security.com/SignServer/adminSig/service/v1.3.0/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:adminsig="http://www.bull.security.com/SignServer/adminSig/v1.3.0/"
">
```

3.1 Dépôt de politique de signature

3.1.1 Description

Cette opération permet à un administrateur de signature de déposer une politique de signature sur le serveur de signature.

Cette politique pourra ensuite être utilisée par un signataire ou une application de signature pour réaliser des opérations de signature ou de vérification de signature.

Cette politique peut être déposée sans qu'elle soit verrouillée; dans ce cas, l'administrateur de signature pourra poser de nouveau la même politique (avec le même OID) qui sera alors mise à jour (le nouveau dépôt remplacera le précédent).

Si la politique a été verrouillée lors de son dépôt alors il ne sera plus possible de la modifier ou la remplacer.

Note : Un document qui serait signé par une politique de signature n'ayant pas été préalablement verrouillée et dont la signature embarque les informations de signature (cas d'une signature EPES) ne pourra plus être vérifiée si la politique a été mise à jour et modifiée (le hash de la politique ne correspondrait alors plus au Hash de la politique embarquée dans la signature).

Nom de l'opération exposé : **depositeSignaturePolicy**

Identifiant SOAP : **depositSigPolicy**

3.1.2 Paramètres d'entrée

Elément depositSigPolicy de type WSDepositSignaturePolicy (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
policyName	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : string 	indique le nom de la politique de signature.
infoFile	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ArtifactInfoFile 	Elément complexe indiquant des informations de description sur la politique de signature(voir descriptif de l'élément au paragraphe 5.9).
lockPolicy	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : boolean 	<p>Indique si la politique doit être verrouillée.</p> <ul style="list-style-type: none"> - True : la politique sera verrouillée et ne sera plus modifiable - False : la politique n'est pas verrouillée et peut donc être modifiée
signaturePolicy	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : signaturePolicy 	Elément complexe contenant la politique de signature au format XML. Cet élément est géré par l'API MetaSIGN et défini dans le document MSIGN-PS-02 .

3.1.3 Paramètres de sortie

Elément artifactIdResponse de type ArtifactIdResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
artifactId	- Attribut - Optionnel - Type : ServerID	indique l'identifiant qui a été associé à la politique de signature déposée.

3.2 Récupération d'une politique de signature

Disponible à partir de la version 1.1.0

3.2.1 Description

Cette opération permet à un administrateur de signature de récupérer une politique de signature précédemment déposée sur le serveur de signature.

Cette politique est récupérée au format XML et correspond au document de description des politique de signature MSIGN-PS-02.

Nom de l'opération exposé : **retrieveSignaturePolicy**

Identifiant SOAP : **retrieveSigPol**

3.2.2 Paramètres d'entrée

Eléménr retrieveSignaturePolicy de type WSRetrieveSignaturePolicy (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signaturePolicyOID	- Attribut - Obligatoire - Type : OID	indique l'OID de la politique de signature à récupérer.

3.2.3 Paramètres de sortie

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signaturePolicy	- Élément - Obligatoire - Type : SignaturePolicy	Elément complexe contenant la politique de signature au format XML. Cet élément est géré défini dans le document MSIGN-PS-02 .

3.3 Suppression d'une politique de signature

Disponible à partir de la version 1.2.0

3.3.1 Description

Cette opération permet à un administrateur de signature de supprimer une politique de signature précédemment déposée sur le serveur de signature.

Nom de l'opération exposé : **deleteSignaturePolicy**

Identifiant SOAP : **deleteSigPol**

3.3.2 Paramètres d'entrée

Élément deleteSigPol de type WSDepleteSignaturePolicy (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signaturePolicyOid	- Attribut - Obligatoire - Type : string	indique l'identifiant OID de la politique de signature qui doit être supprimée

3.3.3 Paramètres de sortie

Élément deleteSigPolResponse de type WSDepleteSignaturePolicyResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signaturePolicyOid	- Attribut - Optionnel - Type : String	indique l'identifiant OID de la politique de signature supprimée

3.4 Récupération d'une liste de politiques de signature présents sur le serveur

Disponible à partir de la version 1.1.2

3.4.1 Description

Récupération d'une liste de politiques de signature répondant à certaines contraintes. La liste des contraintes possibles pour ce type de recherche est donnée en 129. En l'absence de contraintes, l'intégralité des politiques est retournée.

3.4.2 Paramètres d'entrée

Élément listSigPols de type WSListSigPol (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
constraintChain	- Attribut - Optionnel - Type : Chain	Élément complexe qui permet de filtrer la récupération d'une liste (voir descriptif de l'élément au paragraphe 5.17). En l'absence de contrainte, toutes les politiques du serveur seront listées.

3.4.3 Paramètres de sortie

Élément listSigPolsResponse de type ListSigPolResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signaturePolicyInfos	- Élément - Optionnel - Type : SigPolInfo	Élément contenant l'OID de la politique, son nom et son status (verrouillée ou non).

3.5 Dépôt d'un profil de signature

Disponible à partir de la version 1.1.0

3.5.1 Description

Cette opération permet à un administrateur de signature de déposer un profil de signature sur le serveur de signature.

Un profil de signature est identifié par son nom de profil. Lorsque un nouveau profil est déposé avec le nom identique alors il remplace le précédent.

Ce profil de signature pourra ensuite être utilisé par un signataire (rôle « Signer ») ou une application de signature (rôle « SignApplication ») pour réaliser des opérations de signature (ou une augmentation de signature).

Le profil de signature est défini avec les éléments suivants :

- sélection du format de signature parmi:
 - CMS
 - CADES-BES
 - CADES-EPES
 - XADES-BES
 - XADES-EPES
 - PADES-BES
 - PADES-EPES
- sélection du type d'attachement de signature :
 - DETACHED : pour les formats de signature CAdES et XADES,
 - ENVELOPING : pour les formats de signature CAdES et XADES,
 - ENVELOPED : pour les formats de signature XADES et PADES
- sélection d'une politique de signature parmi celles déjà définies
- sélection d'une politique de vérification parmi celles déjà définies
- sélection d'un type d'engagement parmi ceux définis par la politique de signature choisie.
- Le rôle du signataire
- Les algorithmes de signature, de transformation et de canonisation
- optionnellement indique si une date présumé de signature doit être posée
- optionnellement indique si les information sur le lieu de production de signature doit être posé
- optionnellement et dans le cas d'une signature PADES, indiquer si les information sur le contact doivent être précisées.

Optionnellement, l'administrateur a la possibilité d'activer l'option d'archivage et de preuve de la signature.

Nom de l'opération exposé : **depositSignatureProfile**

Identifiant SOAP : **depositSigProfile**

3.5.2 Paramètres d'entrée

Elément depositSigProfile de type WSDepositSignatureProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
infoFile	- Elément - Obligatoire - Type : ArtifactInfoFile	Elément complexe indiquant des informations de description sur le profil de signature(voir descriptif de l'élément au paragraphe 5.9).
signatureProfile	- Elément - Obligatoire - Type : signatureProfileSpecification	Elément complexe contenant les informations sur le profil de signature (voir descriptif de l'élément au paragraphe 5.6.1).

3.5.3 Paramètres de sortie

Elément artifactIdResponse de type ArtifactIdResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.

Identifiant SOAP	Type	Description
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
artifactId	- Attribut - Optionnel - Type : ServerID	indique l'identifiant qui a été associé au profil de signature déposé.

3.6 Mise à jour d'un profil de signature

Disponible à partir de la version 1.1.0

3.6.1 Description

Cette opération permet à un administrateur de signature de mettre à jour un profil de signature sur le serveur de signature.

Un profil de signature est identifié par son identifiant de profil.

Nom de l'opération exposé : **updateSignatureProfile**

Identifiant SOAP : **updateSigProfile**

3.6.2 Paramètres d'entrée

Élément updateSigProfile de type WSUpdateSigProf (extension de WSOperation).

Identifiant SOAP	Propriété	Description
InDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
profileId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant du profil de signature à modifier.

Identifiant SOAP	Propriété	Description
signatureProfile	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : signatureProfileSpecification 	Elément complexe contenant les informations sur le profil de signature (voir descriptif de l'élément au paragraphe 5.6.1).
infoFile	<ul style="list-style-type: none"> - Élément - Optionnel - Type : ArtifactInfoFile 	Elément complexe indiquant des informations de description sur le profil de signature(voir descriptif de l'élément au paragraphe 5.9).

3.6.3 Paramètres de sortie

Elément artifactIdResponse de type ArtifactIdResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
artifactId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerID (string) 	indique l'identifiant qui a été associé au profil de signature déposé. Il doit être identique à celui dans la requête.

3.7 Récupération d'un profil de signature

Disponible à partir de la version 1.1.0

3.7.1 Description

Cette opération permet à un administrateur de signature de récupérer un profil de signature précédemment déposée sur le serveur de signature.

Nom de l'opération exposé : **retrieveSignatureProfile**

Identifiant SOAP : **retrieveSigProf**

3.7.2 Paramètres d'entrée

Elément retrieveSigProf de type WSRetrieveSignatureProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signaturePofileId	- Attribut - Obligatoire - Type : ServerId (string)	indique l'identifiant du profil de signature à récupérer.

3.7.3 Paramètres de sortie

Elément retrieveSigProfResponse de type RetrieveSignatureProfileResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureProfile	- Élément - Optionnel - Type : signatureProfilSpecification	Élément complexe contenant les informations sur le profil de signature (voir descriptif de l'élément au paragraphe 5.6.1).

3.8 Suppression d'un profil de signature

Disponible à partir de la version 1.2.0

3.8.1 Description

Cette opération permet à un administrateur de signature de supprimer un profil de signature précédemment déposé sur le serveur de signature.

Nom de l'opération exposé : **deleteSignatureProfile**

Identifiant SOAP : **deleteSigProfile**

3.8.2 Paramètres d'entrée

Élément deleteSigProf de type WSDeleteSignatureProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureProfileId	- Attribut - Obligatoire - Type : string	indique l'identifiant du profil de signature qui doit être supprimé

3.8.3 Paramètres de sortie

Element deleteSigProfResponse de type DeleteSignatureProfileResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureProfileId	- Attribut - Optionnel - Type : String	indique l'identifiant du profil de signature supprimé

3.9 Récupération d'une liste de profils de signature présents sur le seueur

Disponible à partir de la version 1.2.0

3.9.1 Description

Récupération d'une liste de profils de signature répondant à certaines contraintes. La liste des contraintes possibles pour ce type de recherche est donnée en 130. En l'absence de contraintes, l'intégralité des profils de signature est retournée.

3.9.2 Paramètres d'entrée

Élément *listSigProfile* de type *WSListSigProfile* (extension de *WSOperation*).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type <i>WSOperation</i> au paragraphe 5.1.
constraintChain	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : Chain 	Élément complexe qui permet de filtrer la récupération d'une liste (voir descriptif de l'élément au paragraphe 5.17).

3.9.3 Paramètres de sortie

Élément *listSigProfileResponse* de type *ListSigProfileResponse* (extension de *WSResponse*).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type <i>WSResponse</i> au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type <i>WSResponse</i> au paragraphe 5.2.

sigProfileInfos	<ul style="list-style-type: none"> - Élément - Optionnel - Type : ServerId2NameInfo 	Elément regroupant les attribut du serverID du profil et son nom.
-----------------	--	---

3.10 Dépôt d'un profil de génération de clé de signature

Disponible à partir de la version 1.1.0

3.10.1 Description

Cette opération permet de créer un profil de génération de clé de signature sur le serveur de signature.

Un profil de génération de clé de signature est identifié par son nom de profil. Lorsque un nouveau profil est créé avec le nom identique alors il remplace le précédent.

Ce profil pourra ensuite être utilisé pour la création de bclés de signature et la requête de certificat associé afin de permettre aux applications de signature et signataires d'obtenir un certificat de signature auprès d'une IGC. Ce certificat pourra alors être utilisé pour les opérations de signature.

Le profil de génération de clé de signature est défini avec les éléments suivants :

- la longueur de la clé
- le type de clé : ce type peut être identifié par différents moyens (OID, URI ou chaîne de caractère)

Nom de l'opération exposé : **depositSignatureKeyGenerationProfile**

Identifiant SOAP : **sigKeyGenProfParams**

3.10.2 Paramètres d'entrée

Elément sigKeyGenProfParams de type WSDepSigKeyGenProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOOperation au paragraphe 5.1.
infoFile	- Elément - Obligatoire - Type : ArtifactInfoFile	Elément complexe indiquant des informations de description sur le profil de signature(voir descriptif de l'élément au paragraphe 5.9).
profileContent	- Elément - Obligatoire - Type : SkGenerationProfilSpecification	Elément complexe contenant les informations sur le profil de génération de clé de signature (voir descriptif de l'élément au paragraphe 5.8.2).

3.10.3 Paramètres de sortie

Elément artifactIdResponse de type ArtifactIdResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
artifactId	- Attribut - Optionnel - Type : ServerID	indique l'identifiant qui a été associé au profil de clé de signature déposée.

3.11 Mise à jour d'un profil de génération de clé de signature

Disponible à partir de la version 1.2.0

3.11.1 Description

Cette opération permet à un administrateur de signature de mettre à jour un profil de signature précédemment déposé sur le serveur de signature.

Nom de l'opération exposé : **updateSignatureKeyGenerationProfile**

Identifiant SOAP : **updateSkpProfile**

3.11.2 Paramètres d'entrée

Elément updateSkpProfile de type WSUpdateSkpProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
profileId	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : string 	indique l'identifiant du profil de signature
newProfileContent	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : SkGenerationProfileSpecification 	Élément complexe contenant les informations sur le nouveau profil de génération de clé de signature (voir descriptif de l'élément au paragraphe 5.8.2).

3.11.3 Paramètres de sortie

Elément updateSkgProfileResponse de type WSResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.

3.12 Récupération d'un profil de génération de clé de signature

Disponible à partir de la version 1.1.0

3.12.1 Description

Cette opération permet à un administrateur de signature de récupérer un profil de génération de clé de signature précédemment déposé sur le serveur de signature.

Nom de l'opération exposé : **retrieveSignatureKeyGenerationProfile**

Identifiant SOAP : **retrieveSkgProf**

3.12.2 Paramètres d'entrée

Elément retrieveSkgProf de type WSRetrieveSkgProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signKeyGenPofileId	- Attribut - Obligatoire - Type : ServerId (string)	indique l'identifiant du profil de génération de clé de signature à récupérer.

3.12.3 Paramètres de sortie

Elément retrieveSkgProfResponse de type RetrieveSkgProfileResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureKeyGenerationProfile	- Élément - Optionnel - Type : SkGenerationProfileSpecification	Elément complexe contenant les informations sur le profil de génération de clé de signature (voir descriptif de l'élément au paragraphe 5.8.2).

3.13 Suppression d'un profil de génération de clé de signature

Disponible à partir de la version 1.2.0

3.13.1 Description

Cette opération permet à un administrateur de signature de supprimer un profil de génération de clé de signature précédemment déposé sur le serveur de signature.

Nom de l'opération exposé : **deleteSignatureKeyGenerationProfile**

Identifiant SOAP : **deleteSKGProf**

3.13.2 Paramètres d'entrée

Élément deleteSKGProf de type WSDelSigKeyGenProfile (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
sigKeyGenProfileId	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : string 	indique l'identifiant du profil de génération de clé de signature qui doit être supprimé

3.13.3 Paramètres de sortie

Élément deleteSKGProfResponse de type DeleteSigKeyGenProfileResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
sigKeyGenProfileId	- Attribut - Optionnel - Type : String	indique l'identifiant du profil de génération de clé de signature supprimé

3.14 Récupération d'une liste de profils de génération de clés de signature présents sur le serveur

Disponible à partir de la version 1.2.0

3.14.1 Description

Récupère la liste des profils de génération de clé répondant à certaines contraintes. La liste des contraintes possibles pour ce type de recherche est donnée en 130. En l'absence de contraintes, l'intégralité des profils est retournée.

3.14.2 Paramètres d'entrée

Elément listSKGP de type WSListSKGP (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
constraintChain	- Attribut - Obligatoire - Type : Chain	Élément complexe qui permet de filtrer la récupération d'une liste (voir descriptif de l'élément au paragraphe 116). Si aucune contrainte n'est précisée, l'ensemble des profils sera retourné.

3.14.3 Paramètres de sortie

Élément ISKGPRResponse de type ListSKGPRResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
skgProfileInfos	- Élément - Optionnel - Type : ServerId2NameInfo	Élément associant les attributs de nom et d'identifiant serveur des profils.

3.15 Mise à jour de secret d'activation de clé de signature d'un signataire

Disponible à partir de la version 1.1.0

3.15.1 Description

Cette opération permet à un signataire de modifier le secret d'activation de son certificat et sa clé de signature.

L'appelant fournira alors l'ancien secret d'activation et le nouveau secret d'activation ainsi que son mode d'utilisation.

Nom de l'opération exposé : **updateSignatureSecret**

Identifiant SOAP : **updateSignatureSecret**

3.15.2 Paramètres d'entrée

Elément updateSignatureSecret de type WSUpdateSignatureSecret (extension de WSUsignatureOperation et WSUpdateSignatureSecret).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ActivationSecret 	Elément complexe contenant l'ancien secret d'activation de la clé de signature (voir descriptif de l'élément au paragraphe 5.11).
newSignatureSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ActivationSecret 	Elément complexe contenant le nouveau secret d'activation de la clé de signature (voir descriptif de l'élément au paragraphe 5.11).

3.15.3 Paramètres de sortie

Elément updateSignatureSecretResponse de type WSResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.

3.16 Dépôt d'un container de clé PKCS#12

Disponible à partir de la version 1.1.0

3.16.1 Description

Cette opération permet à un administrateur de déposer un certificat de signature et la clé privée associée au serveur de signature sous forme de PKCS#12 encodé avec un mot de passe de transport.

Ce PKCS#12 sera ensuite associé à un signataire et un secret d'activation pour permettre au signataire de réaliser des signature grâce à la clé de signature et le certificat associé.

Nom de l'opération exposé : **depositKeyStore**

Identifiant SOAP : **depositKeystore**

3.16.2 Paramètres d'entrée

Élément depositKeystore de type WSDepositKeystore (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.

Identifiant SOAP	Propriété	Description
keystore	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Keystore 	Elément complexe contenant les informations sur le PKCS#12 (voir descriptif de l'élément au paragraphe 5.10).

3.16.3 Paramètres de sortie

Elément depositKeystoreResponse de type DepositKeystoreResponse (extension de WsResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
keystoreId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	indique l'identifiant qui a été associé au container PKCS#12 déposé.

3.17 Création d'une clé de signature à partir d'un PKCS#12

Disponible à partir de la version 1.1.0

3.17.1 Description

Cette opération permet à un administrateur d'associer un signataire à un container PKCS#12 contenant une clé de signature et son certificat de signature associé.

L'association ne pourra se faire que si l'opération contient le bon mot de passe de transport du PKCS#12 à associer et que le secret d'activation correspond bien au signataire. De plus, si les usages de clé pour le certificat sont renseignés, l'opération ne pourra être effectuée que si les usages de clés

du certificat contenu dans le PKCS#12 sont conformes à ceux demandés.

A la fin de cette opération, le signataire pourra utiliser cette clé de signature pour signer des documents.

Nom de l'opération exposé : **createSignatureKeyFromPKCS12**

Identifiant SOAP : **createSkcFPkcs12**

3.17.2 Paramètres d'entrée

Elément createSkcFPkcs12 de type WSCreateSignatureKeyCertFromPkcs12 (extension de WSSignatureOperation et WSOOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOOperation au paragraphe 5.1.
signatureSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ActivationSecret 	Elément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).
SignatureKeyId	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : SignatureKeyId 	Identifiant de la clé de signature à créer. Pour un même utilisateur les identifiants de clé sont uniques.
p12TransportPassword	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : PassphraseActivationSecret 	Elément complexe contenant le mot de passe de transport du PKCS#12 référencé dans la requête (voir descriptif de l'élément au paragraphe 5.11.1).
p12Id	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId (string) 	indique l'identifiant PKCS#12 préalablement déposé et devant être associé au signataire.

Identifiant SOAP	Propriété	Description
keyUsage	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : KeyUsageArray 	Indique les usages de clés qui doivent être présents pour le certificat (§5.22).

3.17.3 Paramètres de sortie

Élément createSkcResponse de type CreateSignatureKeyCertResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureKeyId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : SignatureKeyId 	Paramètre contenant l'identifiant de la clé de signature qui a été associée à la création de la clé de signature. Il doit être identique à celui transmis dans les paramètres d'entrée.

3.18 Création d'une clé de signature à partir d'un identifiant de clé dans le HSM

Disponible à partir de la version 1.1.0

3.18.1 Description

Cette opération permet à un administrateur définir une clé de signature à partir de l'identifiant d'une paire de clés (clé privée/clé publique) générée au sein du HSM (Hardware Secure Module) lors d'une procédure tiers (ex : cérémonie de clé).

L'association ne pourra se faire si l'opération contient un identifiant de clé connu par le HSM et non déjà déclaré dans le serveur de signature comme clé de signature.

A la fin de cette opération, le signataire pourra utiliser cette clé de signature pour faire une demande de certificat de signature. Si les usages de clés sont définis, le certificat de signature généré à partir de cette clé de signature aura ces usages de clés.

Ce mode est préconisé lorsque la clé de signature est pour un usage de clé cachet serveur.

Nom de l'opération exposé : **createSignatureKeyCertFromKeyPair**

Identifiant SOAP : **createSkcFKP**

3.18.2 Paramètres d'entrée

Elément createSkcFKP de type WSCreateSignatureKeyCertFromKeyPair (extension de WSSignatureOperation et WSOOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOOperation au paragraphe 5.1.
signatureSecret	- Élément - Obligatoire - Type : ActivationSecret	Elément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).
keyPairCkald	- Attribut - Obligatoire - Type : base64Binary	indique l'identifiant CKaID de la clé générée dans le HSM et devant être associé au signataire.
signatureKeyId	- Attribut - Obligatoire - Type : SignatureKeyId	Paramètre contenant l'identifiant de la clé de signature qui sera associée à la création de la clé de signature.
keyUsage	- Attribut - Optionnel - Type : KeyUsageArray	Indique les usages de clés qui doivent être présents pour le certificat (§5.22).

3.18.3 Paramètres de sortie

Elément createSkcResponse de type CreateSignatureKeyCertResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureKeyId	- Attribut - Optionnel - Type : SignatureKeyId (string)	Paramètre contenant l'identifiant de la clé de signature qui a été associée à la création de la clé de signature. Il doit être identique à celui transmis dans les paramètres d'entrée.

3.19 Création d'une clé de signature à partir d'un profil de clé de signature

Disponible à partir de la version 1.1.0

3.19.1 Description

Cette opération permet à un administrateur définir une clé de signature à partir d'un profil de signature préalablement déposé.

L'association ne pourra se faire que si l'opération contient un identifiant de profil connu par le serveur de signature ou une spécification explicite d'un nouveau profil de génération de clé de signature.

A la fin de cette opération, le signataire pourra utiliser cette clé de signature pour faire une demande de certificat de signature. Si les usages de clés sont définis, le certificat de signature généré à partir de cette clé de signature aura ces usages de clés.

Nom de l'opération exposé : **createSignatureKeyFromProfile**

Identifiant SOAP : **createSkcFProfile**

3.19.2 Paramètres d'entrée

Elément createSkcFProfile de type WSCreateSignatureKeyCertFromProfile (extension de

WSSignatureOperation et WSOperation).

Identifiant SOAP		Propriété	Description
inDelegationOf		- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureSecret		- Élément - Obligatoire - Type : ActivationSecret	Élément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).
signatureKeyId		- Attribut - Obligatoire - Type : SignatureKeyId (string)	Paramètre contenant l'identifiant de la clé de signature qui sera associée à la création de la clé de signature.
Au choix	skgProfileId	- Élément - Optionnelle - Type : SkGenerationProfileReference	Indique l'identifiant du profil de génération de clé de signature préalablement déposé
	signatureKeyGenerationProfile	- Élément - Optionnelle - Type : SkGenerationProfileSpecification	Élément complexe contenant les informations sur le profil de génération de clé de signature (voir descriptif de l'élément au paragraphe 5.8.2).

3.19.3 Paramètres de sortie

Élément createSkcResponse de type CreateSignatureKeyCertResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureKeyID	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : SignatureKeyld (string) 	Paramètre contenant l'identifiant de la clé de signature qui a été associée à la création de la clé de signature. Il doit être identique à celui transmis dans les paramètres d'entrée.

3.20 Suppression d'une clé de signature

Disponible à partir de la version 1.2.0

3.20.1 Description

Cette opération permet à un administrateur de signature de supprimer une clé de signature précédemment déposée sur le serveur de signature.

Nom de l'opération exposé : **deleteSignatureKey**

Identifiant SOAP : **deleteSigKey**

3.20.2 Paramètres d'entrée

Elément deleteSigKey de type WSDestroySignatureKey (extension de WSSignatureOperation et WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureKeyId	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : SignatureKeyId (string) 	Paramètre contenant l'identifiant de la clé de signature qui doit être supprimée
signatureSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ActivationSecret 	Élément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).

3.20.3 Paramètres de sortie

Élément deleteSigKeyResponse de type DeleteSignatureKeyResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
signatureKeyId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Paramètre contenant l'identifiant de la clé de signature qui a été supprimée. Il doit être identique à celui transmis dans les paramètres d'entrée.

3.21 Création d'une requête de certificat pour une clé de signature existante

Disponible à partir de la version 1.1.0

3.21.1 Description

Cette opération permet de demander la génération d'une requête de certification (fichier CSR) pour une clé de signature au serveur de signature conformément aux usages de clés définis pour cette clé. Le fichier CSR permet ensuite de faire une demande de certificat auprès d'une IGC pour en obtenir un certificat de signature. Ce certificat sera ensuite importé sur le serveur par une autre opération de Service Web.

Le fichier CSR permet ensuite de faire une demande de certificat auprès d'une IGC pour en obtenir un certificat de signature. Ce certificat sera ensuite importé sur le serveur par une autre opération de Service Web.

Le fichier CSR ne sera créé que si le secret d'activation correspond au signataire.

Nom de l'opération exposée : **generateCSRForSignatureKey**

Identifiant SOAP : **generateCsrForSkc**

3.21.2 Paramètres d'entrée

Elément generateCsrForSKC de type WSGenerateCsrForSignatureKey (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ActivationSecret 	Elément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).

Identifiant SOAP	Propriété	Description
signatureKeyId	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : SignatureKeyId 	Paramètre contenant l'identifiant de la clé de signature qui sera associée à la création de la clé de signature.

3.21.3 Paramètres de sortie

Élément csrForSKCResponse de type CsrForSignatureKeyResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
signaturekeyId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : SignatureKeyId (string) 	indique l'identifiant de la clé de signature qui a été générée.
publickey	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : base64Binary 	contient la clé publique générée et encodée en Base 64.
csr	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : base64Binary 	contient la requête de certificat associée à la clé générée et encodée en Base 64.

3.22 Dépôt d'un certificat de signature pour une clé de signature existante

Disponible à partir de la version 1.1.0

3.22.1 Description

Cette opération permet à un administrateur de déposer un certificat de signature associée à une clé de signature qui a été préalablement générée ou déposée sur le serveur de signature.

Le certificat de signature ne sera associée à la clé de signature que si le secret d'activation correspond au signataire.

Nom de l'opération exposé : **depositCertificateForSignatureKey**

Identifiant SOAP : **depositCertForSkc**

3.22.2 Paramètres d'entrée

Elément depositCertForSkc de type WSDepositCertForSignatureKey (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureSecret	- Élément - Obligatoire - Type : ActivationSecret	Elément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).
signaturekeyId	- Attribut - Obligatoire - Type : SignatureKeyId	indique l'identifiant de la clé de signature qui doit être associé au certificat déposé.
x509Certificate	- Attribut - Obligatoire - Type : base64Binary	contient le certificat de signature à déposer et encodée en Base 64.

3.22.3 Paramètres de sortie

Elément depositCertForSkcResponse de type WSResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.

3.23 Activation d'une clé de signature

Disponible à partir de la version 1.4.0

3.23.1 Description

Cette opération permet à un signataire (directement ou via délégation à une de ses applications de confiance) de demander l'activation d'une de ses clés de signature.

Cette opération ne s'applique qu'aux clés générées à partir d'un profil faisant intervenir une authentification renforcée (ex. DASAAuthentication). Elle est alors nécessaire en préalable à toute utilisation de la clé pour

- la signature d'un document (voir 14),
- la génération d'une CSR (voir 55),
- ou l'import d'un certificat de signature (voir 57).

L'opération d'activation retourne un challenge (donnée générée possiblement par le HSM) qui permettra à l'utilisateur de déterminer le secret d'activation dynamique (DAS) à envoyer lors de l'utilisation de la clé pour l'opération protégée souhaitée (signature de document, génération de CSR, import de certificat de signature).

Cas de l'authentification avec OTP :

Si un service de gestion d'OTP (broker OTP) a été configuré dans l'entité Signature-Server-Config, l'opération retourne un résultat succès sans le challenge. Dans ce cas, le challenge est envoyé au broker qui a la charge d'envoyer l'OTP à l'utilisateur.

Cas de l'authentification avec FIDO U2F :

Dans ce cas, l'opération retourne le challenge ainsi que les paramètres nécessaires à l'authentification FIDO.

Nom de l'opération exposée : **activateSignatureKey**

Identifiant SOAP : **activateSignatureKey**

3.23.2 Paramètres d'entrée

Elément activateSignatureKey de type WSActivateSignatureKey(extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureKeyId	- Élément - Obligatoire - Type : SignatureKeyId	Identifiant de la clé de signature à activer (voir 125).
activationData	- Élément - Obligatoire - Type : ActivationData	Elément complexe permettant d'activer la clé(voir descriptif de l'élément au paragraphe 5.19).

3.23.3 Paramètres de sortie

Elément activateSignatureKeyResponse de type ActivateSignatureKeyResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.

Identifiant SOAP	Type	Description
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
activationResult	- Attribut - Optionnel - Type : ActivationResult	Données d'activation (voir 122).

3.24 Génération de certificat de signature pour une clé de signature existante

Disponible à partir de la version 1.4.0

3.24.1 Description

Cette opération permet d'envoyer une requête de génération de certificat pour une clé de signature existante. Le serveur de signature contacte une IGC afin d'effectuer la demande de certificat de signature en transmettant une CSR générée automatiquement par le serveur de signature conformément aux usages de clés définis pour la clé de signature.

Cette opération repose sur la configuration du serveur de signature dans laquelle est renseignée :

- Les paramètres de connexion permettant d'atteindre le fournisseur de certificats ;
- Les paramètres inhérents au protocole utilisé (SCEP, A2M, CMP...).

Le fournisseur de certificat est contacté que si la clé de signature a été autorisée à l'utilisation par le signataire.

Nom de l'opération exposée : **requestCertificateForSignatureKey**

Identifiant SOAP : **requestCertForSkc**

3.24.2 Paramètres d'entrée

Elément requestCertForSkc de type WSRequestCertificateForSignatureKey (extension de WSSignatureOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signatureSecret	- Élément - Obligatoire - Type : ActivationSecret	Élément complexe contenant le secret d'activation du signataire (voir descriptif de l'élément au paragraphe 5.11).
signaturekeyId	- Attribut - Obligatoire - Type : SignatureKeyId	indique l'identifiant de la clé de signature qui doit être associé au certificat déposé.
certRequestParameters	- Élément - Obligatoire - Type : CertificateRequestParameters	Élément complexe abstrait contenant les paramètres de la requête de demande de certificat de signature (voir descriptif de l'élément au paragraphe 5.24).

3.24.3 Paramètres de sortie

Élément requestCertForSkcResponse de type CertificateForSignatureKeyResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.

Identifiant SOAP	Type	Description
signaturekeyId	<ul style="list-style-type: none">- Attribut- Optionnel- Type : SignatureKeyId (string)	indique l'identifiant de la clé de signature qui a été utilisée pour l'opération.
certResult	<ul style="list-style-type: none">- Element- Optionnel- Type : RequestCertificateResult	Contient la réponse du fournisseur de certificat de signature. (voir descriptif de l'élément au paragraphe 5.25).

→ 4.23

4 Service Web : Administration des utilisateurs du serveur de signature

Ce chapitre décrit toutes les opérations exposées par le services Web d'administration des utilisateurs du serveur de signature de signature.

Elles permettent à un administrateur ou une application d'administration de réaliser les opérations de gestion des utilisateurs du serveur.

Elles permettent à un administrateur ou une application d'administration de :

- Créer, mettre à jour ou supprimer des utilisateurs;
- Créer ou supprimer des groupes;
- Ajouter ou retirer des utilisateurs dans un groupe;
- Consulter une liste de groupes existant;
- Consulter une liste d'utilisateurs existants ;
- Gérer les token FIDO pour des utilisateurs.
- URL du Service :

<https://hostname:port/servlets/SignServer/com.bull.security.coreserver.ws.admin/services/AdminSOAP>

« Hostname » est le nom du serveur (ou adresse IP) sur lequel le serveur de signature MetaSIGN-Serveur est déployé

« Port » est le numéro de port écouté par le serveur.

- Espaces de nommages :

```
targetNamespace="http://www.bull.security.com/Server/admin/service/v1.3.0/"
xmlns:wSDL="http://schemas.xmlsoap.org/wSDL/"
xmlns:tns="http://www.bull.security.com/Server/admin/service/v1.3.0/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wSDL/soap/"
xmlns:admin="http://www.bull.security.com/Server/admin/v1.3.0/">
```

4.1 Création d'un utilisateur

4.1.1 Description

Cette opération permet à un administrateur de créer et déclarer un utilisateur du serveur de signature.

La déclaration d'un utilisateur s'accompagne des informations suivantes :

- le nom de l'utilisateur permettant de l'identifier;
- le ou les modes d'authentification de l'utilisateur (par mot de passe, ou certificat);
- le ou les rôles dont il disposera;
- le ou les groupes auxquels il appartient.
- Le ou les autres utilisateurs (applications) pour lesquels l'utilisateur donne délégation.

Nom de l'opération exposé : **createUser**

Identifiant SOAP : **createUser**

4.1.2 Paramètres d'entrée

Élément createUser de type WSCreateUser (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
parameter	- Élément - Obligatoire - Type : UserSpecification	Élément complexe indiquant les informations de description de l'utilisateur (voir descriptif de l'élément au paragraphe 5.12.2).

4.1.3 Paramètres de sortie

Élément createUserResponse de type CreateUserResponse (extension de Wsresponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
userId	- Attribut - Optionnel - Type : ServerId	indique l'identifiant qui a été associé à la création de l'utilisateur.

4.2 Mise à jour d'un utilisateur

Disponible à partir de la version 1.1.0

4.2.1 Description

Cette opération permet à un administrateur (ou à l'utilisateur lui même) de mettre à jour les informations d'un utilisateur du serveur de signature.

La mise à jour permet de modifier les informations suivantes :

- le nom de l'utilisateur permettant de l'identifier;
- le ou les modes d'authentification de l'utilisateur (par mot de passe, ou certificat);
- le ou les rôles dont il disposera;
- le ou les groupes auxquels il appartient.
- Le ou les applications auxquelles l'utilisateur donne délégation.

Nom de l'opération exposé : **updateUser**

Identifiant SOAP : **updateUser**

4.2.2 Paramètres d'entrée

Elément updateUser de type WSUpdateUser (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOOperation au paragraphe 5.1.
userId	- Attribut - Obligatoire - Type : ServerId (string)	indique l'identifiant de l'utilisateur à modifier
newUser	- Élément - Obligatoire - Type : User	Nouvelles informations de description de l'utilisateur (voir descriptif de l'élément au paragraphe 5.12.2). Attention seuls les éléments de type UserSpecification ou ApplicationSpecification seront pris en compte.

Notes sur l'utilisation du newUser : pour les Rôles, Credentials, Groupes et Applications lorsque l'on souhaite

- conserver les éléments existants
 - il faut utiliser les <Élément>Reference correspondants (sans y inclure de spécifications)
- retirer des éléments existants
 - il suffit de ne pas les faire apparaître dans la requête
- créer de nouveaux éléments ou remplacer des éléments existants par de nouveaux
 - il faut utiliser les <Élément>Specification correspondants

Les <Élément>Reference sont renvoyés dans le résultat de l'opération ListUsers (voir 79). Le plus aisé pour mettre à jour un utilisateur est donc de récupérer sa description au moyen d'un ListUser et de la modifier selon ce qui est souhaité pour constituer le newUser désiré.

4.2.3 Paramètres de sortie

Élément updateUserResponse de type updateUserResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
updatedId	- Attribut - Optionnel - Type : ServerId	indique l'identifiant qui a été associé à la mise à jour de l'utilisateur. (il doit être identique à celui donné en paramètre d'entrée)

4.2.4 Exemples

Les exemples suivants sont applicables à la version 1.1.2 du serveur de signature.

Ils se basent sur la mise à jour d'un utilisateur ayant les caractéristiques suivantes :

- id « 38 »
- nom « Test Signer »
- authentifié par certificat X509
- de rôle « Signer »
- ayant pour application de confiance l'application d'identifiant « 4 »

Pour obtenir les informations de cet utilisateur, une requête « listUser » (dont la contrainte IdList ne contient que l'id « 38 ») est appelé et donne le résultat suivant :

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns4:listUsersResponse xmlns:ns7="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
xmlns:ns6="http://www.bull.com/metasign/xmlSignaturePolicy/v3.1#"
xmlns:ns5="http://www.bull.security.com/Server/coreServices/v1.1.0/"
xmlns:ns4="http://www.bull.security.com/Server/admin/v1.1.0/" xmlns:ns3="http://www.bull.security.com/Server/jobsMgt/v1.1.0/"
xmlns:ns2="http://www.quartz-scheduler.org/xml/JobSchedulingData" returnStatus="MSIGN_SRV_STATUS_SUCCESS">
      <ns4:users xmlns:ns9="http://www.bull.security.com/Server/coreAdmin/v1.1.0/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="ns9:UserReference" userId="38">
        <value name="Test Signer">
          <roles xsi:type="ns9:RoleReference" userId="38"
roleNS="http://www.bull.security.com/SignServer/extensions/v1.1.0/#Signer"/>
          <credentials xsi:type="ns9:CredentialReference">
            <credentialId userId="38">
<credentialKey>x509Certificate_vrtdyV1XNSFLgcs/WdftlUskGVM=</credentialKey>
            </credentialId>
          </credentials>
          <groups xsi:type="ns9:GroupReference" groupId="43246"/>
          <trustedApplications xsi:type="ns9:ApplicationReference" applicationId="4"/>
        </value>
      </ns4:users>
    </ns4:listUsersResponse>
  </soap:Body>
</soap:Envelope>
```

4.2.4.1 Mise à jour du nom d'un utilisateur

La requête ci-dessous permet la mise à jour du nom de l'utilisateur d'identifiant « 38 » en « Test Signer Updated ».

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:updateUser xmlns:ns4="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
xmlns:ns3="http://www.bull.security.com/Server/coreServices/v1.1.0/"
xmlns:ns2="http://www.bull.security.com/Server/admin/v1.1.0/" userId="38">
      <ns2:newUser xmlns:ns6="http://www.bull.security.com/Server/coreAdmin/v1.1.0/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="ns6:UserSpecification" name="Test Signer Updated">
        <roles xsi:type="ns6:RoleReference" userId="38"
roleNS="http://www.bull.security.com/SignServer/extensions/v1.1.0/#Signer"/>
          <credentials xsi:type="ns6:CredentialReference">
            <credentialId userId="38">
<credentialKey>x509Certificate_vrtdyV1XNSFLgcs/WdftlUskGVM=</credentialKey>
              </credentialId>
            </credentials>
            <groups xsi:type="ns6:GroupReference" groupId="43246"/>
            <trustedApplications xsi:type="ns6:ApplicationReference" applicationId="4"/>
          </ns2:newUser>
        </ns2:updateUser>
      </soap:Body>
    </soap:Envelope>
```

4.2.4.2 Ajout d'une authentification par mot de passe

La requête ci-dessous permet d'ajouter une authentification par mot de passe à l'utilisateur dont l'identifiant est « 38 »

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:updateUser xmlns:ns4="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
xmlns:ns3="http://www.bull.security.com/Server/coreServices/v1.1.0/"
xmlns:ns2="http://www.bull.security.com/Server/admin/v1.1.0/" userId="38">
      <ns2:newUser xmlns:ns6="http://www.bull.security.com/Server/coreAdmin/v1.1.0/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="ns6:UserSpecification" name="Test Signer">
        <roles xsi:type="ns6:RoleReference" userId="38"
roleNS="http://www.bull.security.com/SignServer/extensions/v1.1.0/#Signer"/>
          <credentials xsi:type="ns6:CredentialReference">
            <credentialId userId="38">
              <credentialKey>x509Certificate_vrtdyV1XNSFLgcs/WdftlUskGVM=</credentialKey>
            </credentialId>
          </credentials>
          <credentials xsi:type="ns6:PasswordWrapper"
hashAlgo="SHA-256" passwordHash="XohlMNoobHFR0OVjcYpJ3NgPQ1qq73WKHvch0VQtg="/>
            <groups xsi:type="ns6:GroupReference" groupId="43246"/>
            <trustedApplications xsi:type="ns6:ApplicationReference" applicationId="4"/>
          </ns2:newUser>
        </ns2:updateUser>
      </soap:Body>
    </soap:Envelope>
```

4.2.4.3 Changement de l'application de confiance

La requête ci-dessous permet de modifier l'application de confiance associé à l'utilisateur dont l'identifiant est « 38 ». L'application de confiance est identifié par l'identifiant « 20 » (qui doit exister sur le serveur).

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:updateUser xmlns:ns4="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
xmlns:ns3="http://www.bull.security.com/Server/coreServices/v1.1.0/"
xmlns:ns2="http://www.bull.security.com/Server/admin/v1.1.0/" userId="38">
      <ns2:newUser xmlns:ns6="http://www.bull.security.com/Server/coreAdmin/v1.1.0/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="ns6:UserSpecification" name="Test Signer">
        <roles xsi:type="ns6:RoleReference" userId="38"
roleNS="http://www.bull.security.com/SignServer/extensions/v1.1.0/#Signer"/>
          <credentials xsi:type="ns6:CredentialReference">
            <credentialId userId="38">
<credentialKey>x509Certificate_vrtdyV1XNSFLgcs/WdftlUskGVM=</credentialKey>
              </credentialId>
            </credentials>
            <groups xsi:type="ns6:GroupReference" groupId="43246"/>
              <trustedApplications xsi:type="ns6:ApplicationReference" applicationId="20"/>
            </ns2:newUser>
          </ns2:updateUser>
        </soap:Body>
      </soap:Envelope>
```

4.3 Supprimer un utilisateur

Disponible à partir de la version 1.1.0

4.3.1 Description

Cette opération permet de supprimer un utilisateur du serveur de signature.

La suppression d'un utilisateur n'est autorisé que par :

- un application ayant le rôle de « ServerManager »
- un utilisateur ayant les droits d'administrateur sur le serveur de signature
- un administrateur appartenant au groupe où l'utilisateur est affecté.

Nom de l'opération exposé : **deleteUser**

Identifiant SOAP : **deleteUser**

4.3.2 Paramètres d'entrée

Elément deleteUser de type WSDelateUser (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
userId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant de l'utilisateur à supprimer

4.3.3 Paramètres de sortie

Elément deleteUserResponse de type DeleteResponse (extension de WSOperation).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.

Identifiant SOAP	Type	Description
deletedId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	indique l'identifiant de l'utilisateur qui a été supprimé. (il doit être identique à celui donné en paramètre d'entrée)

4.4 Créer un groupe

Disponible à partir de la version 1.1.0

4.4.1 Description

Cette opération permet à un administrateur de créer et déclarer un groupe dans lequel pourra être affectés des utilisateurs et/ou d'autres groupes du serveur de signature.

La déclaration d'un groupe s'accompagne des informations suivantes :

- le nom du groupe permettant de l'identifier;
- une liste d'identifiants de groupes existants associés à ce groupe ;
- une liste d'identifiants d'utilisateurs associés à ce groupe;

Nom de l'opération exposé : **createGroup**

Identifiant SOAP : **createGroup**

4.4.2 Paramètres d'entrée

Elément createGroup de type WSCreateGroup (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.
parameter	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : GroupSpecification 	Elément complexe indiquant les informations de description du groupe(voir descriptif de l'élément au paragraphe 5.15.2).

4.4.3 Paramètres de sortie

Elément createGroupResponse de type WSCreateGroupResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
groupId	- Attribut - Optionnel - Type : ServerId	indique l'identifiant qui a été associé à la création du groupe.

4.5 Supprimer un groupe

Disponible à partir de la version 1.1.0

4.5.1 Description

Cette opération permet de supprimer un groupe du serveur de signature.

La suppression d'un utilisateur n'est autorisé que lorsque ce groupe ne contient plus d'utilisateur. Pour ce faire l'administrateur aura au préalable retirer les utilisateurs du groupe.

Cette opération est autorisé pour :

- un application ayant le rôle de « ServerManager »
- un utilisateur ayant les droits d'administrateur sur le serveur de signature

Nom de l'opération exposé : **deleteGroup**

Identifiant SOAP : **deleteGroup**

4.5.2 Paramètres d'entrée

Elément deleteGroup de type WSDeleteGroup (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
groupId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant du groupe à supprimer

4.5.3 Paramètres de sortie

Elément deleteGroupResponse de type DeleteResponse (WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
deletedId	- Attribut - Optionnel - Type : ServerId (string)	indique l'identifiant du groupe qui a été supprimé. (il doit être identique à celui donné en paramètre d'entrée)

4.6 Ajouter des utilisateurs dans un groupe

Disponible à partir de la version 1.1.0

4.6.1 Description

Cette opération permet d'ajouter un ou plusieurs utilisateurs à un groupe.

L'opération est réalisée par une association groupe/utilisateur(s).

Une association groupe/utilisateur(s) est composé de :

- l'identification d'un groupe
- un ou plusieurs identifiant(s) d'utilisateur(s)

Les utilisateurs alors associés au groupe peuvent utilisés les artéfacts (politiques de signature, profils de signature, profils de clé) inclus dans ce groupe lors des différents opérations sur le serveur de signature.

Si, au moins l'un des identifiants des utilisateurs n'est pas connu du serveur de signature, alors l'opération d'affectation des utilisateurs dans le groupe ne sera pas réalisée. L'administrateur devra donc recommencer l'opération avec les bons identifiants d'utilisateurs.

Cette opération est autorisé pour :

- un application ayant le rôle de « ServerManager »
- un utilisateur ayant les droits d'administrateur sur le serveur de signature

Nom de l'opération exposé : **addUsersToGroup**

Identifiant SOAP : **addUsersToGroup**

4.6.2 Paramètres d'entrée

Elément addUsersToGroup de type WSAddUsersToGroup (extension de WSOperation).

Identifiant SOAP	Propriété	Description
------------------	-----------	-------------

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOOperation au paragraphe 5.1.
parameter	- Élément - Obligatoire - Type : UsersToGroupAssociation	Élément complexe qui renvoi les informations sur une association groupe/utilisateur(s).(voir descriptif de l'élément au paragraphe 5.16).

4.6.3 Paramètres de sortie

Élément addUsersToGroupResponse de type UsersGroupResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
groupId	- Attribut - Optionnel - Type : ServerId (string)	indique l'identifiant du groupe dont les utilisateurs ont été associés (il doit être identique à celui donné en paramètre d'entrée)

4.7 Retirer des utilisateurs dans un groupe

Disponible à partir de la version 1.1.0

4.7.1 Description

Cette opération permet de retirer un ou plusieurs utilisateurs d'un groupe.

L'opération est réalisée par une association groupe/utilisateur(s).

Une association groupe/utilisateur(s) est composé de :

- l'identification d'un groupe
- un ou plusieurs identifiant(s) d'utilisateur(s)

Les utilisateurs sont alors retirés du groupe et ne pourront plus utiliser les artéfacts (politiques de signature, profils de signature, profils de clé) inclus dans ce groupe lors des différents opérations sur le serveur de signature.

Si, au moins l'un des identifiants de utilisateurs n'est pas connu du groupe, alors l'opération de retrait des utilisateurs du groupe ne sera pas réalisée. L'administrateur devra donc recommencer l'opération avec les bons identifiants d'utilisateurs.

Cette opération est autorisé pour :

- un application ayant le rôle de « ServerManager »
- un utilisateur ayant les droits d'administrateur sur le serveur de signature

Nom de l'opération exposé : **removeUsersFromGroup**

Identifiant SOAP : **removeUsersFromGroup**

4.7.2 Paramètres d'entrée

Elément removeUsersFromGroup de type WSRemoveUsersFromGroup (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.

Identifiant SOAP	Propriété	Description
parameter	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : UsersToGroupAssociation 	Élément complexe qui renvoi les informations sur une association groupe/utilisateur(s).(voir descriptif de l'élément au paragraphe 5.16).

4.7.3 Paramètres de sortie

Élément removeUsersFromGroupResponse de type UsersFromGroupResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
groupId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	indique l'identifiant du groupe dont les utilisateurs ont été retirés (il doit être identique à celui donné en paramètre d'entrée)

4.8 Consulter une liste de groupes existant

Disponible à partir de la version 1.1.0

Récupération de la liste des groupes répondants à certains critères. La liste des contraintes possibles pour ce type de recherche est donnée en 129.

4.8.1 Description

Nom de l'opération exposé : **listGroups**

Identifiant SOAP : **listGroups**

4.8.2 Paramètres d'entrée

Élément *listGroups* de type *WSListGroups* (extension de *WSOperation*).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type <i>WSOperation</i> au paragraphe 5.1.
templateGroup	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : GroupSpecification 	<p>Élément décrivant un groupe d'exemple servant de critère de recherche.</p> <p>ATTENTION : cette fonctionnalité n'est pas encore implémentée ; utiliser les contraintes à la place via <i>constraintChain</i>.</p>
constraintChain	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Chain 	Élément complexe qui permet de filtrer la récupération d'une liste (voir descriptif de l'élément au paragraphe).

4.8.3 Paramètres de sortie

Élément *listGroupsResponse* de type *ListGroupsResponse* (extension de *WSResponse*).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
groups	<ul style="list-style-type: none"> - Élément - Optionnel - Type : GroupReference 	Voir descriptif du type GroupReference au paragraphe 115.
totalResutls	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : Integer 	Nombre total de réponses (si celui-ci ne correspond pas à celui retourné dans la réponse).

4.8.4 Exemple

L'exemple suivant est applicables à la version 1.1.2 du serveur de signature.

La requête ci-dessous permet de faire une recherche des groupes dont le nom contient « Admins » et l'identifiant du groupe est compris entre 1 et 50000

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ns4="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
xmlns:ns3="http://www.bull.security.com/Server/coreServices/v1.1.0/"
xmlns:ns2="http://www.bull.security.com/Server/admin/v1.1.0/">
  <soap:Body>
    <ns2:listGroups>
      <ns2:constraintChain>
        <ns3:constraint xsi:type="ns3:StringConstraint"
          type="CONTAINS" value="Admins"
          constrainedElement="http://www.bull.security.com/Server/coreAdmin/v1.1.0/#Group.name"/>
        <ns3:links>
          <ns3:booleanOperator>AND</ns3:booleanOperator>
          <ns3:constraint xsi:type="ns3:IdRange"
            from="1"
            to="50000"
            constrainedElement="http://www.bull.security.com/Server/coreAdmin/v1.1.0/#Group.id"/>
        </ns3:links>
      </ns2:constraintChain>
    </ns2:listGroups>
  </soap:Body>
</soap:Envelope>
```

La réponse à cette requête donne les informations suivantes sur le groupe qui a été trouvé :

- Identifiant du groupe « 43246 »,
- Nom du groupe « Admins1386764607763 »
- Identifiants des utilisateur dans ce groupe « 4 » et « 38 ».

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ns9="http://www.bull.security.com/Server/coreAdmin/v1.1.0/"
xmlns:ns7="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
xmlns:ns6="http://www.bull.com/metasign/xmlSignaturePolicy/v3.1#"
xmlns:ns5="http://www.bull.security.com/Server/coreServices/v1.1.0/"
xmlns:ns4="http://www.bull.security.com/Server/admin/v1.1.0/"
xmlns:ns3="http://www.bull.security.com/Server/jobsMgt/v1.1.0/"
xmlns:ns2="http://www.quartz-scheduler.org/xml/JobSchedulingData">
  <soap:Body>
    <ns4:listGroupsResponse returnStatus="MSIGN_SRV_STATUS_SUCCESS">
      <ns4:groups groupId="43246">
        <value name="Admins1386764607763">
          <ns9:users xsi:type="ns9:UserReference" userId="4"/>
          <ns9:users xsi:type="ns9:UserReference" userId="38"/>
        </value>
      </ns4:groups>
    </ns4:listGroupsResponse>
  </soap:Body>
</soap:Envelope>
```

4.9 Consulter une liste d'utilisateurs existants

Disponible à partir de la version 1.1.0

Récupération de la liste des utilisateurs répondants à certains critères. La liste des contraintes possibles pour ce type de recherche est donnée en 129.

4.9.1 Description

Nom de l'opération exposé : **listUsers**

Identifiant SOAP : **listUsers**

4.9.2 Paramètres d'entrée

Élément *listUsers* de type *WSListUsers* (extension de *WSOperation*).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type <i>WSOperation</i> au paragraphe 5.1.
constraintChain	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : Chain 	Élément complexe qui permet de filtrer la récupération d'une liste (voir descriptif de l'élément au paragraphe).
userTypeFilter	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : UserTypeEnum 	Cet attribut permet de filtrer la requête en fonction du type d'utilisateur. Exemple : si userTypeFilter= « com.bull.security.server.core.Applicatio », alors seuls les utilisateurs de type application seront retournés.

4.9.3 Paramètres de sortie

Élément *listUsersResponse* de type *ListUsersResponse* (extension de *WSResponse*).

Identifiant SOAP	Type	Description
------------------	------	-------------

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
users	- Élément - Optionnel - Type : User	Utilisateurs correspondant aux critères de recherche. Ne sont renvoyées que des UserSpecification.
totalResutls	- Attribut - Optionnel - Type : Integer	Nombre total de réponses (si celui-ci ne correspond pas à celui retourné dans la réponse).

4.9.4 Exemple

L'exemple suivant est applicables à la version 1.1.2 du serveur de signature.

La requête ci-dessous permet de faire une recherche des utilisateurs dont le nom contient «Test» et l'identifiant du groupe est compris entre 1 et 100 ou dont le nom contient « Admin »

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sigsrvexts="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
  xmlns:coreservices="http://www.bull.security.com/Server/coreServices/v1.1.0/"
  xmlns:admin="http://www.bull.security.com/Server/admin/v1.1.0/">

  <soap:Body>
    <admin:listUsers>
      <admin:constraintChain>
        <coreservices:constraint xsi:type="coreservices:StringConstraint"
          type="CONTAINS" value="Test"
          constrainedElement="http://www.bull.security.com/Server/coreAdmin/v1.1.0/#User.name"/>
        <coreservices:links>
          <coreservices:booleanOperator>AND</coreservices:booleanOperator>
          <coreservices:constraint xsi:type="coreservices:IdRange"
            from="1"
            to="100"
            constrainedElement="http://www.bull.security.com/Server/coreAdmin/v1.1.0/#User.id"/>
        </coreservices:links>
        <coreservices:links>
          <coreservices:booleanOperator>OR</coreservices:booleanOperator>
          <coreservices:constraint xsi:type="coreservices:StringConstraint"
            type="CONTAINS" value="Admin"
            constrainedElement="http://www.bull.security.com/Server/coreAdmin/v1.1.0/#User.name"/>
        </coreservices:links>
      </admin:constraintChain>
    </admin:listUsers>
  </soap:Body>
</soap:Envelope>
```

La réponse à cette requête donne les informations suivantes sur les utilisateurs qui ont été trouvés :

- un Utilisateur :
 - Identifiant « 38 »
 - Nom « Test signer »
 - Rôle « Signer »
 - Authentification par certificat X509
 - Identifiant de l'application de confiance associé « 4 »

- une Application :
 - Identifiant « 4 »
 - Nnom « Super_Admin_Application »
 - Rôles « SignManager » et « ServerManager »
 - Authentification par certificat X509 ou mot de passe.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:ns9="http://www.bull.security.com/Server/coreAdmin/v1.1.0/"
    xmlns:ns7="http://www.bull.security.com/SignServer/extensions/v1.1.0/"
    xmlns:ns5="http://www.bull.security.com/Server/coreServices/v1.1.0/"
    xmlns:ns4="http://www.bull.security.com/Server/admin/v1.1.0/"
    xmlns:ns3="http://www.bull.security.com/Server/jobsMgt/v1.1.0/"
    xmlns:ns2="http://www.quartz-scheduler.org/xml/JobSchedulingData">

  <soap:Body>

    <ns4:listUsersResponse returnStatus="MSIGN_SRV_STATUS_SUCCESS" >
      <ns4:users xsi:type="ns9:UserReference" userId="38">
        <value name="Test Signer">
          <roles xsi:type="ns9:RoleReference" userId="38"
            roleNS="http://www.bull.security.com/SignServer/extensions/v1.1.0/#Signer"/>
          <credentials xsi:type="ns9:CredentialReference">
            <credentialId userId="38">
              <credentialKey>x509Certificate_vrtdyV1XNSFLgcs/WdftlUskGVM=</credentialKey>
            </credentialId>
          </credentials>
          <trustedApplications xsi:type="ns9:ApplicationReference" applicationId="4"/>
        </value>
      </ns4:users>
      <ns4:users xsi:type="ns9:ApplicationReference" applicationId="4">
        <value name="Super_Admin_Application">
          <roles xsi:type="ns9:RoleReference" userId="4"
            roleNS="http://www.bull.security.com/SignServer/extensions/v1.1.0/#SignManager"/>
          <roles xsi:type="ns9:RoleReference" userId="4"
            roleNS="http://www.bull.security.com/Server/coreAdmin/v1.1.0/#ServerManager"/>
          <credentials xsi:type="ns9:CredentialReference">
            <credentialId userId="4">
              <credentialKey>password</credentialKey>
            </credentialId>
          </credentials>
        </value>
      </ns4:users>
    </ns4:listUsersResponse>
  </soap:Body>
</soap:Envelope>
```

```

        </credentialId>
    </credentials>
    <credentials xsi:type="ns9:CredentialReference">
        <credentialId userId="4">
            <credentialKey>x509Certificate_2pzSBI2avn+VNks7gKObx3Q+Fzg=</credentialKey>
        </credentialId>
    </credentials>
</value>
</ns4:users>
</ns4:listUsersResponse>
</soap:Body>
</soap:Envelope>

```

4.10 Consulter un utilisateur

Disponible à partir de la version 1.3.0

Récupération des caractéristiques d'un utilisateur.

4.10.1 Description

Nom de l'opération exposé : **consultUser**

Identifiant SOAP : **consultUser**

4.10.2 Paramètres d'entrée

Elément **consultUser** de type **WSConsultUser** (extension de **WSOperation**).

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Voir descriptif du type WSOperation au paragraphe 5.1.

Identifiant SOAP	Propriété	Description
id	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ServerId 	Identifiant de l'utilisateur à consulter.

4.10.3 Paramètres de sortie

Elément consultUserResponse de type ConsultUserResponse (extension de WSResponse).

Identifiant SOAP	Type	Description
returnStatusEnum	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Voir descriptif du type WSResponse au paragraphe 5.2.
name	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Nom de l'utilisateur
roles	<ul style="list-style-type: none"> - Élément - Optionnel - Type : RoleDescription 	Moyens d'authentification de l'utilisateur voir descriptif du type au §4.10.4
credentials	<ul style="list-style-type: none"> - Élément - Optionnel - Type : CredentialDescription 	Descriptions des moyens d'authentification de l'utilisateur voir descriptif du type au §4.10.5
groups	<ul style="list-style-type: none"> - Élément - Optionnel - Type : ServerId2NameInfo 	Groupes de l'utilisateur voir descriptif du type au §123

Identifiant SOAP	Type	Description
trustedApplications	<ul style="list-style-type: none"> - Élément - Optionnel - Type : ServerId2NameInfo 	Applications de confiance de l'utilisateur voir descriptif du type au §123
certificatesInfo	<ul style="list-style-type: none"> - Élément - Optionnel - Type : tableau de CertificateInfo 	Informations sur les certificats de signature dont dispose l'utilisateur

4.10.4 RoleDescription

Description du rôle d'un utilisateur.

Identifiant SOAP	Type	Description
roleNS	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : String 	Identifiant du rôle sous la forme : {nameSpace}#{nom du Role}

4.10.5 CredentialDescription

Description d'un moyen d'authentification d'un utilisateur.

Identifiant SOAP	Type	Description
credentialKey	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : String 	Chaine de caractères identifiant pour un utilisateur donné de façon unique un mode d'authentification existant.

Identifiant SOAP	Type	Description
password	- Élément - Optionnel - Type : String	Elément vide dont la présence indique l'existence d'un mot de passe.
X509Certificate	- Élément - Optionnel - Type : Base64String	Représentation encodée en BASE64 du certificat.

4.10.6 CertificateInformation

Description des informations du certificat.

Identifiant	Propriété	Description
certificate	- Élément - Optionnel - Type : Base64String	Certificat (encodé en base64)
authMethod	- Attribut - Optionnel - Type : String	Méthode d'authentification associée au certificat (ex : DAS_BASIC_OTP...)

4.10.7 Exemple

L'exemple suivant est applicables à la version 1.3.0 du serveur de signature.

La réponse ci-dessous est le résultat d'une requête « consultUser » dont les rôles sont ServerManager et SignManager, capable de s'authentifier avec un mot de passe ou l'un des deux certificats retournés (leur représentation base 64 a été tronquée dans cette documentation).

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

  <soap:Body>

    <ns3:consultUserResponse xmlns:ns7="http://www.bull.security.com/SignServer/extensions/v1.3.0/"
xmlns:ns6="http://www.bull.com/metasign/xmlSignaturePolicy/v3.1#"
xmlns:ns5="http://www.bull.security.com/Server/coreServices/v1.3.0/" xmlns:ns4="http://www.quartz-
scheduler.org/xml/JobSchedulingData" xmlns:ns3="http://www.bull.security.com/Server/admin/v1.3.0/"
xmlns:ns2="http://www.bull.security.com/Server/jobsMgt/v1.3.0/" name="Default response"
returnStatus="MSIGN_SRV_STATUS_SUCCESS">

      <ns3:roles roleNS="http://www.bull.security.com/Server/coreAdmin/v1.3.0/#ServerManager"/>

      <ns3:roles roleNS="http://www.bull.security.com/SignServer/extensions/v1.3.0/#SignManager"/>

      <ns3:credentials credentialKey="x509Certificate_...">

        <ns3:x509Certificate>MIIFajCCA1KgAwIBAg...+vFHI8VMDHlw==</ns3:x509Certificate>

      </ns3:credentials>

      <ns3:credentials credentialKey="x509Certificate_...">

        <ns3:x509Certificate>MIIFcTCCA1mgAwIBAgI ...kBCdHAg3Q=</ns3:x509Certificate>

      </ns3:credentials>

      <ns3:credentials credentialKey="password">

        <ns3:password/>

      </ns3:credentials>

      <ns3:groups serverId="20" name="Default group 20"/>

      <ns3:groups serverId="21" name="Default group 21"/>

      <ns3:groups serverId="22" name="Default group 22"/>

      <ns3:groups serverId="23" name="Default group 23"/>

      <ns3:groups serverId="24" name="Default group 24"/>

      <ns3:trustedApplications serverId="1" name="Default trusted App 1"/>

      <ns3:trustedApplications serverId="2" name="Default trusted App 2"/>

    </ns3:consultUserResponse>

  </soap:Body>

</soap:Envelope>
```

4.11 Initialisation du processus d' enrôlement FIDO

Disponible à partir de la version 2.4.0

4.11.1 Description

Cette opération permet d'envoyer une requête d'initialisation du processus d' enrôlement d' un token FIDO pour un utilisateur.

Cette opération repose sur la configuration du serveur de signature dans laquelle est renseignée :

- Les paramètres liés aux tokens FIDO.

Nom de l'opération exposée : **enrollFIDOInit**

Identifiant SOAP : **enrollFIDOInit**

4.11.2 Paramètres d'entrée

Elément enrollFIDOInit de type WSEnrollFIDOInit (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signerId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant de l'utilisateur (signataire) à enrôler.

4.11.3 Paramètres de sortie

Elément enrollFIDOInitResponse de type EnrollFIDOInitResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
appld	- Attribut - Optionnel - Type : String	indique l'identifiant du serveur de signature.
challenge	- Element - Optionnel - Type : Base64String	Contient le challenge en base 64.
registeredTokens	- Element - Optionnel - Type : Array of RegisteredToken	Tableau des tokens enregistrés. Voir §5.28
version	- Attribut - Optionnel - Type : String	Version du protocole U2F utilisé (ex :U2F_V2)

4.12 Dépôt d'un token FIDO pour un utilisateur

Disponible à partir de la version 2.4.0

4.12.1 Description

Cette opération permet d'envoyer une requête de dépôt d'un token FIDO pour un utilisateur.

Cette opération repose sur la configuration du serveur de signature dans laquelle est renseignée :

- Les paramètres liés aux tokens FIDO.

Nom de l'opération exposée : **depositFIDOToken**

Identifiant SOAP : **depositFIDOToken**

4.12.2 Paramètres d'entrée

Élément depositFIDOToken de type WSDepositFIDOToken (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOperation au paragraphe 5.1.
signerId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant de l'utilisateur (signataire) propriétaire du token FIDO.
registrationData	- Attribut - Optionnel - Type : base64Binary	Tableau de réponses encodé en base64 [pubKey, keyHandle, attestation certificate, signature]
clientData	- Attribut - Optionnel - Type : base64Binary	Données client encodées en base64 [navigator.id.finishEnrollment, challenge, cid_pubKey, appId]

4.12.3 Paramètres de sortie

Élément depositFIDOTokenResponse de type DepositFIDOTokenResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.

Identifiant SOAP	Type	Description
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
tokenId	- Attribut - Optionnel - Type : ServerId	indique l'identifiant du token FIDO enregistré.

4.13 Récupération des identifiants des token FIDO

Disponible à partir de la version 2.4.0

4.13.1 Description

Cette opération permet d'envoyer une requête de récupération des token FIDO pour un utilisateur.

Cette opération repose sur la configuration du serveur de signature dans laquelle est renseignée :

- Les paramètres liés aux tokens FIDO.

Nom de l'opération exposée : **listFIDOToken**

Identifiant SOAP : **listFIDOToken**

4.13.2 Paramètres d'entrée

Élément listFIDOToken de type WSListFIDOToken (extension de WSOOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOOperation au paragraphe 5.1.
signerId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant de l'utilisateur (signataire) dont on veut récupérer les token FIDO.

4.13.3 Paramètres de sortie

Elément listFIDOTokenResponse de type ListFIDOTokenResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
tokenIdList	- Attribut - Optionnel - Type : ServerId List	Liste des identifiants des token FIDO

4.14 Suppression d'un token FIDO

Disponible à partir de la version 2.4.0

4.14.1 Description

Cette opération permet d'envoyer une requête de supprimer un token FIDO.

Cette opération repose sur la configuration du serveur de signature dans laquelle est renseignée :

- Les paramètres liés aux tokens FIDO.

Nom de l'opération exposée : **deleteFIDOToken**

Identifiant SOAP : **deleteFIDOToken**

4.14.2 Paramètres d'entrée

Elément deleteFIDOToken de type WSDeleteFIDOToken (extension de WSOperation).

Identifiant SOAP	Propriété	Description
inDelegationOf	- Attribut - Optionnel - Type : ServerId	Voir descriptif du type WSOOperation au paragraphe 5.1.
tokenId	- Attribut - Obligatoire - Type : ServerId	indique l'identifiant du token FIDO à supprimer.

4.14.3 Paramètres de sortie

Élément deleteFIDOTokenResponse de type DeleteFIDOTokenResponse.

Identifiant SOAP	Type	Description
returnStatusEnum	- Attribut - Obligatoire - Type : ReturnStatusEnum	Voir descriptif du type WSResponse au paragraphe 5.2.
errorInfo	- Attribut - Optionnel - Type : String	Voir descriptif du type WSResponse au paragraphe 5.2.
fidoToken	- Attribut - Optionnel - Type : ServerId	Identifiant du token supprimé

5 Définition des éléments complexes des paramètres des Web services

Ce chapitre décrit les différentes structures complexes utilisées par les services Web du serveur de signature.

Toutes les opérations effectuées via les Web-Services sont envoyées au sein d'éléments étendant l'élément « WSOperation ». La réponse renvoyée par le serveur est contenue dans un élément étendant l'élément « WSResponse ». Ces deux types définissent respectivement l'ensemble des caractéristiques communes à toutes les opérations ou réponses des Web Services du serveur de signature.

5.1 Type « WSOperation »

Type abstrait regroupant les caractéristiques communes d'une opération du Web Service. Il permet de gérer les aspects liés à la délégation des opérations.

Identifiant SOAP	Propriété	Description
inDelegationOf	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Lorsque l'on souhaite effectuer une opération en délégation, ce paramètre permettant de préciser l'identifiant de l'utilisateur pour le compte duquel l'opération sera réalisée.

5.2 Type « WSResponse »

Type abstrait regroupant les caractéristiques communes d'une réponse du Web Service. Il permet de gérer les aspects au statu de retour du serveur et à un éventuel message d'erreur.

Identifiant SOAP	Propriété	Description
returnStatus	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ReturnStatusEnum 	Renvoi le statut de la requête Valeurs possibles: <ul style="list-style-type: none"> - MSIGN_SRV_STATUS_SUCCESS - MSIGN_SRV_STATUS_INTERNAL_ERROR - MSIGN_SRV_STATUS_REJECTED_REQUEST
errorInfo	<ul style="list-style-type: none"> - Élément - Optionnel - Type : string 	Renvoi une information complémentaire lors d'une erreur d'exécution de la requête. Cette information est optionnelle.

5.3 Type « Document »

Les éléments de ce type représentent un document envoyé via l'une des opérations :

- dépôt de document WSDepDocSig
- demande de signature WSSignDocument
- demande de vérification de signature WSVerifSig

Un document peut avoir au choix l'une des natures suivantes :

1. contenu (le contenu du document est alors effectivement envoyé lors de l'opération, encodé en base64)
2. référence à un document déjà déposé sur le serveur (l'identifiant de document est alors envoyé lors de l'opération)
3. référence à une URL (cette URL ainsi que les informations nécessaires à son traitement, type mime et encodage sont alors envoyées lors de l'opération)
4. hash du document (ceci n'est valable que pour les opérations de signature ou de vérification de signatures détachées durant lesquelles seul le hash du document et non son contenu seront transmis au serveur)

Identifiant		Propriété	Description
Au choix	docID	- Élément - Obligatoire - Type : ServerId	Identifiant du document. Celui-ci a été préalablement déposé sur le serveur ou généré par celui-ci (cas du signature).
	docURL	- Élément - Obligatoire - Type : anyURI	Lien URL (HTTP) permettant d'accéder au document.
	docContent	- Élément - Obligatoire - Type : base64Binary	Contenu du document. ce contenu doit être encodé en « Base64 » et présenté en « US-ASCII »
	docHash	- Élément - Obligatoire - Type : string	contient le Hash du document. Cette méthode est uniquement valable pour le document original ne contenant pas la signature. Cette méthode peut être utilisée lorsqu'un document est « trop volumineux » ou « trop sensible » pour être transmis directement au serveur.

Identifiant	Propriété	Description
contentType	- Attribut - Optionnel - Type : MimeType (string)	Définit le type MIME du document. Ce paramètre est requis lorsque le document est envoyé par la méthode d'accès « URL ». Il sera ignoré dans les autres cas.
encoding	- Attribut - Optionnel - Type : anyURI	Définit l'encodage du contenu. Ce paramètre est requis lorsque le document est envoyé par la méthode d'accès « URL ». Il sera ignoré dans les autres cas.

5.4 Type « SignerCertificate »

Identifiant	Propriété	Description
dbIdentifier	- Élément - Obligatoire - Type : ServerId	Élément contenant l'identifiant de l'utilisateur.
x509Certificate	- Élément - Obligatoire - Type : base64Binary	Élément contenant le certificat X509 encodé en Base64

5.5 Type « SignVerifReport »

Identifiant	Propriété	Description
1 à n fois SignatureReport	- Element - Obligatoire - Type : SignatureReport	Élément complexe contenant le rapport de vérification de la signature. (voir descriptif de l'élément au paragraphe 5.5.1).

Identifiant	Propriété	Description
status	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	<p>Indique le statut de la vérification de la signature.</p> <ul style="list-style-type: none"> - True : la signature est correctement vérifiée et le rapport est correct. - False : la vérification de la signature a échoué car au moins un élément dans le rapport est incorrect.

5.5.1 Type SignatureReport

Identifiant	Propriété	Description
signatureNumber	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : string 	Indique l'indice de la signature qui a été vérifiée. (dans le cas où le document dispose de plusieurs signatures).
report	<ul style="list-style-type: none"> - Element - Obligatoire - Type : rapportType 	Elément complexe contenant le rapport de vérification de la signature. Cet élément est géré par l'API MetaSIGN et défini dans le document MSIGN-API-GDE-03.

5.6 Type « SignatureProfile »

Identifiant	Propriété	Description
Au choix	<p>profID</p> <ul style="list-style-type: none"> - Element - Obligatoire - Type : ServerId 	Elément contenant l'identifiant d'un profil de signature.

Identifiant	Propriété	Description
profile	<ul style="list-style-type: none"> - Element - Obligatoire - Type : SignatureProfileSpecification 	Elément complexe contenant les informations sur le profil de signature (voir descriptif de l'élément au paragraphe 5.6.1).

5.6.1 Type SignatureProfileSpecification

Identifiant	Propriété	Description
signaturePolicyOID	<ul style="list-style-type: none"> - Element - Optionnel - Type : string 	Elément contenant l'OID d'une politique de signature.
verificationPolicyOID	<ul style="list-style-type: none"> - Element - Optionnel - Type : string 	Elément contenant l'OID d'une politique de signature utilisée pour la vérification de la signature. Si celle-ci est absente, alors la politique identifiée dans la signature sera utilisée.
attachment	<ul style="list-style-type: none"> - Element - Obligatoire - Type : AttachmentType (string) 	<p>Elément indiquant les informations sur le type d'attachement de la signature.</p> <p>Les valeurs possibles sont :</p> <ul style="list-style-type: none"> - DETACHED - ENVELOPING - ENVELOPED

Identifiant		Propriété	Description
format		<ul style="list-style-type: none"> - Element - Obligatoire - Type : SignatureFormat 	<p>Elément indiquant le format de la signature.</p> <p>Les valeurs possibles sont :</p> <ul style="list-style-type: none"> - CMS - CADES_BES - CADES_EPES - XADES_BES - XADES_EPES - PADES_BES - PADES_EPES
choix entre l'un ou l'autre	augmentation	<ul style="list-style-type: none"> - Element - Obligatoire - Type : AugmentationType 	<p>Elément indiquant le format d'augmentation de la signature.</p> <p>Les valeurs possibles sont :</p> <ul style="list-style-type: none"> - NONE - T - C - X1 - X2- XL - A - LTV
	augmentationLevel	<ul style="list-style-type: none"> - Element - Obligatoire - Type : AugmentationLevelType 	<p>Elément indiquant le niveau d'augmentation de la signature.</p> <p>Les valeurs possibles sont :</p> <ul style="list-style-type: none"> - B - T - LT - LTA

Identifiant		Propriété	Description
1 à n fois	commitments	<ul style="list-style-type: none"> - Element - Optionnel - Type : CommitmentType 	Elément indiquant une liste autorisée des types d'engagement pour la signature.
	signerRole	<ul style="list-style-type: none"> - Element - Optionnel - Type : SignerRole 	Elément complexe contenant une description du rôle du signataire. Il est composée de l'identification de l'organisation et le rôle du signataire dans cette organisation.
	signatureAlgoId	<ul style="list-style-type: none"> - Element - Optionnel - Type : string 	Elément indiquant l'identifiant de l'algorithme de signature.
	transformationAlgo	<ul style="list-style-type: none"> - Element - Optionnel - Type : TransformationType 	Elément complexe contenant une description des algorithmes de transformation qui seront appliqués
	canonicalizationAlgo	<ul style="list-style-type: none"> - Element - Optionnel - Type : CanonicalizationAlgo 	Elément complexe contenant l'URI de l'algorithme de canonicalisation qui sera appliqué.
	requireSigningTime	<ul style="list-style-type: none"> - Element - Optionnel - Type : boolean 	Indique si une information sur la date supposée de la signature doit être présent dans la signature
	requirePlaceOfSignature	<ul style="list-style-type: none"> - Element - Optionnel - Type : boolean 	Indique si les informations sur la localisation de la signature doivent être présentes dans la signature
	requireContactInfo	<ul style="list-style-type: none"> - Element - Optionnel - Type : boolean 	Indique si les informations de contact du signataire doivent être présentes dans la signature (uniquement en cas de signature PAdES)
	archive	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	Indique si la signature doit être archivée dans le système d'archivage par le serveur de signature.

Identifiant	Propriété	Description
name	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : string 	Indique le nom du profil de signature.

5.7 Type « SignatureOptionalInfos »

Identifiant SOAP		Propriété	Description
deActivateAutoTS		<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	Désactive l'apposition automatique d'un jeton d'horodatage dans une signature PAdES simple
placeOfSignature		<ul style="list-style-type: none"> - Élément - Optionnel - Type : SignatureProductionPlaceType 	Élément complexe indiquant des informations sur le lieu où est réalisée la signature (voir descriptif de l'élément au paragraphe 5.7.1).
1 à n fois	commitments	<ul style="list-style-type: none"> - Element - Optionnel - Type : CommitmentType (string) 	Élément indiquant une liste autorisée des types d'engagement pour la signature.
contactInfo		<ul style="list-style-type: none"> - Élément - Optionnel - Type : string 	Élément indiquant une information sur le contact du signataire.
visualSignature		<ul style="list-style-type: none"> - Élément - Optionnel - Type : VisualSignature 	Élément complexe indiquant des informations sur la signature visuelle (voir descriptif de l'élément au paragraphe 102). Cet élément ne sera utilisé que dans le cadre d'une signature PAdES.

5.7.1 Type SignatureProductionPlaceType

Identifiant	Propriété	Description
City	- attribut - Optionnel - Type : string	Attribut contenant la ville.
StateOrProvince	- attribut - Optionnel - Type : string	Attribut contenant l'état ou la province.
PostalCode	- attribut - Optionnel - Type : string	Attribut contenant le code postal.
CountryName	- attribut - Optionnel - Type : string	Attribut contenant le pays.

5.7.2 Type VisualSignature

Éléments nécessaires à l'apposition d'une signature visuelle (dans un document PDF). Ce paramètre est silencieusement ignoré lors des augmentations ou lors des signatures qui ne sont pas au format PAdES.

Il s'agit d'un type abstrait se déclinant en : VisualSignatureFromFieldName ou VisualSignatureFromPosition. Chaque élément permet de définir une image de fonds, une image de la signature manuscrite ainsi que le positionnement de la zone de signature au sein du document PDF à signer. Les images de fonds ou de signature manuscrites sont automatiquement retaillées pour rentrer dans la zone de signature si nécessaire.

Identifiant	Propriété	Description
logoBackground	- élément - Optionnel - Type : Document	Image du « watermark » qui se trouve en fond de la signature visuelle sur le document signé.

Identifiant	Propriété	Description
logoLoyalty	<ul style="list-style-type: none"> - élément - Optionnel - Type : Document 	Image de représentation de la signature manuscrite à placer sur le document signé.
font	<ul style="list-style-type: none"> - élément - Optionnel - Type : string 	Définit la police du texte qui sera affiché dans la signature visuelle. Cette police doit appartenir à la liste des polices utilisables définies par le système sur lequel le serveur de signature est exécuté
certificateObjectInfo	<ul style="list-style-type: none"> - élément - Optionnel - Type : string 	<p>Définit les éléments récupérés du « Distinguished Name » (DN) de l'objet du certificat.</p> <p>Si l'utilisateur souhaite afficher le DN en entier alors il ne doit pas renseigner cette information.</p> <p>L'affichage des éléments se fera selon l'ordre indiqué dans la fonction.</p> <p>Un contrôle syntaxique par expression régulière (Majuscule et/ou minuscule) est réalisé.</p>
displayableElement	<ul style="list-style-type: none"> - élément (0 à n) - Optionnel - Type : VisualSignatureDisplayableElement 	<p>Définit un élément à afficher (cf voir les implémentations de</p> <p>VisualSignatureDisplayableElement).</p>

5.7.2.1 Type VisualSignatureFromFieldName

Les éléments nécessaires au positionnement de la zone de signature visuelle dans une page de document PDF sont définis à partir d'une zone prédéfinies dans le document PDF (cas de formulaire) et désignée par un nom de champ.

Identifiant	Propriété	Description
-------------	-----------	-------------

Identifiant	Propriété	Description
signatureFieldName	<ul style="list-style-type: none"> - attribut - Optionnel - Type : string 	Nom du champ dans le document PDF représentant l'encadrement de la signature visuelle dans lequel seront positionnées les informations de la signature visuelle.

5.7.2.2 Type VisualSignatureFromPosition

Les éléments nécessaires au positionnement de la zone de signature visuelle dans une page de document PDF sont définis à partir d'une page et d'éléments de position. Les unités pour les éléments de position ou dimension sont le pixel du document à signer.

Identifiant	Propriété	Description
page	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : Entier 	<p>Numéro de la page sur laquelle la signature visible sera insérée selon la nomenclature suivante :</p> <p>En partant de la première page :</p> <p style="padding-left: 40px;">0 : première page</p> <p style="padding-left: 40px;">1 : deuxième page</p> <p style="padding-left: 40px;">Etc.</p> <p>En partant de la dernière page :</p> <p style="padding-left: 40px;">-1 : dernière page</p> <p style="padding-left: 40px;">-2 : avant-dernière page</p> <p style="padding-left: 40px;">Etc</p>
xAxis	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : Entier 	Distance du rectangle par rapport au bord gauche de la page dans laquelle la signature doit être insérée.
yAxis	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : Entier 	Distance du rectangle par rapport au bas de la page dans laquelle la signature doit être insérée
signatureHeight	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : Entier 	Hauteur de l'encadré de la signature visuelle dans le document.

Identifiant	Propriété	Description
signatureWidth	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : Entier 	Largeur de l'encadré de la signature visuelle dans le document.

5.7.3 Type VisualSignatureDisplayableElement

Type abstrait permettant de définir un élément affichable dans la signature.

Identifiant	Propriété	Description
text	<ul style="list-style-type: none"> - élément - Optionnel - Type : String 	Définit le texte de l'élément à afficher.

5.7.3.1 Type VisualSignatureAliasElt

Élément permettant de définir le texte d'information précédant l'alias (nom convivial) associé au certificat dans son conteneur. La taille maximale du texte est de 40 caractères.

5.7.3.2 Type VisualSignatureLocationElt

Élément permettant de définir le texte d'information précédant la localisation (si précisé dans la signature). La taille maximale du texte est de 40 caractères.

5.7.3.3 Type VisualSignatureCommitmentTypeElt

Élément permettant de définir le texte d'information précédant le type d'engagement (si précisé dans la signature). La taille maximale du texte est de 40 caractères.

5.7.3.4 Type VisualSignatureDateElt

Élément permettant de définir le texte d'information précédant la date de signature. Le format de la date affichée sera JJ/MM/AAAA (AAAA/MM/JJ en langue anglaise). La taille maximale du texte est de 40 caractères.

5.7.3.5 Type VisualSignatureDNElt

L'élément définit le texte qui précède l'affichage du DN ou d'une partie du DN du sujet du certificat. La taille maximale du texte est de 40 caractères.

5.8 Type « SkGenerationProfile »

Type abstrait décliné en SkGenerationProfileReference et SkGenerationProfileSpecification.

5.8.1 Type SkGenerationProfileReference

Identifiant	Propriété	Description
skgProfileId	<ul style="list-style-type: none"> - attribut - Optionnel - Type : ServerId 	Attribut contenant l'identifiant d'un profil de clé de signature.

5.8.2 Type SkGenerationProfileSpecification

Identifiant	Propriété	Description
kpAlgo	<ul style="list-style-type: none"> - Element - Obligatoire - Type : KPType 	Elément complexe contenant une description du type de clé représenté sous différentes forme. (voir descriptif de l'élément au paragraphe 5.8.3). L'utilisateur a la possibilité de choisir l'une de ces forme représentatives.
algoParameters	<ul style="list-style-type: none"> - Element - Optionnel - Type : AlgoParameters 	Elément complexe destiné à recevoir les paramètres de définition de l'algorithme de création de clé (voir la description en 106).
enforcedAuthentication	<ul style="list-style-type: none"> - Element - Optionnel - Type : EnforcedAuthenticationMethod 	Elément complexe destiné à recevoir les paramètres d'authentification renforcée (voir la description en 119).

Identifiant	Propriété	Description
keyUsage	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : KeyUsageArray 	Indique les usages de clés qui doivent être présents pour le certificat (§5.22).

5.8.3 Type KPTYPE

Identifiant	Propriété	Description
oid	<ul style="list-style-type: none"> - Element - Obligatoire - Type : KPTYPE 	Elément contenant le type de clé représenté sous forme d'un OID.
uri	<ul style="list-style-type: none"> - Element - Obligatoire - Type : QName 	Elément contenant le type de clé représenté sous forme d'une URI.
name	<ul style="list-style-type: none"> - Element - Obligatoire - Type : string 	Elément contenant le type de clé représenté sous forme d'une chaîne de caractères (ex : RSA).

5.8.4 Type AlgoParameters

Identifiant	Propriété	Description
parameter	<ul style="list-style-type: none"> - Element - Optionnel - Type : Parameter 	Elément contenant la paire clé valeur correspondant à un paramètre.

5.8.4.1 Type Parameters

Identifiant	Propriété	Description
key	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : string 	clé identifiant le paramètre.
value	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : string 	Valeur du paramètre.

5.9 Type « ArtifactInfoFile »

Identifiant		Propriété	Description
1 à n fois	description	<ul style="list-style-type: none"> - Element - Obligatoire - Type : DescriptionEntry 	Elément complexe contenant une description de l'artefact. (voir descriptif de l'élément au paragraphe 5.9.1).
	keyWords	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : KeywordsEntry 	Elément complexe contenant des mots clé identifiant l'artefact. (voir descriptif de l'élément au paragraphe 5.9.2).

5.9.1 Type DescriptionEntry

Identifiant	Propriété	Description
lang	<ul style="list-style-type: none"> - Element - Obligatoire - Type : string 	Elément contenant la langue utilisée (exemple :FR ou EN).

Identifiant	Propriété	Description
description	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : string 	Elément contenant la description dans la langue positionnée dans l'élément précédent.

5.9.2 Type KeywordsEntry

Identifiant		Propriété	Description
lang		- Elément - Obligatoire - Type : string	Elément contenant la langue utilisée (exemple :FR ou EN).
1 à n fois	keyWords	- Attribut - Obligatoire - Type : string	Elément contenant un mot clé dans la langue positionnée dans l'élément précédent.

5.10 Type « Keystore »

Type abstrait se déclinant pour l'instant uniquement en en Pkcs12Keystore.

5.10.1 Type Pkcs12Keystore

Identifiant	Propriété	Description
content	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : base64Binary 	<p>Attribut contenant le PKCS#12 encodé en base 64.</p> <p>Note : pour des données issues de fichiers PEM exclure les lignes marquant le début et la fin du P12.</p>

5.11 Type « ActivationSecret »

Type abstrait qui protège l'accès aux informations contenues dans les coffres forts de signature des signataires par un secret. Il se décline soit en PassPhraseActivationSecret, DASActivationSecret ou FIDOActivationSecret.

5.11.1 Type PassphraseActivationSecret

Identifiant	Propriété	Description
bytes	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : base64Binary 	Élément contenant un mot de passe d'activation encodé en base 64.

5.11.2 DASActivationSecret

Il étend PassPhraseActivationSecret en lui adjoignant un DAS dédié à l'utilisation de chaque clé du coffre fort. Son utilisation est nécessaire pour utiliser toute clé créée avec un profil de génération comprenant une authentification renforcée (par ex. DASAAuthentication) et doit faire suite à l'activation préalable de la clé (via l'opération ActivateSignatureKey du WS SigOps).

Identifiant	Propriété	Description
dynamicSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : base64Binary 	Secret d'activation dynamique (DAS) encodé en base 64.

5.11.3 FIDOActivationSecret

Il étend PassPhraseActivationSecret en lui adjoignant des données FIDO permettant l'utilisation de la clé du coffre-fort si l'authentification est effectuée à l'aide de ces données. Son utilisation est nécessaire

pour utiliser toute clé créée avec un profil de génération comprenant une authentification renforcée (par ex. FIDOAuthentication) et doit faire suite à une action d'initialisation pour l'enrôlement de l'utilisateur propriétaire de la clé FIDO (via l'opération EnrollFIDOInit du WS SigOps).

Identifiant	Propriété	Description
signatureData	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : base64Binary 	Tableau de réponses encodé en base64 [user presence, counter, signature]
clientData	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : base64Binary 	Données client encodées en base64 [navigator.id.finishEnrollment, challenge, cid_pubKey, applId]
fidoToken	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : string 	Identifiant du token FIDO associé à l'utilisateur

5.12 Type « User »

Type abstrait regroupant les différentes représentations d'un utilisateur. Il se décline en UserReference (référence à un utilisateur déjà existant sur le serveur) ou UserSpecification (spécification d'un utilisateur non existant sur le serveur).

5.12.1 Type UserReference

Décrit un utilisateur déjà existant sur le serveur. Il est renvoyé par les opérations de listage d'utilisateurs et envoyé pour les opérations de mises à jour.

Identifiant	Propriété	Description
userId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Attribut contenant l'identifiant de l'utilisateur.

Identifiant	Propriété	Description
value	-Élément -Optionnel -Type : UserSpecification	Description des caractéristiques de l'utilisateur (ou de sa mise à jour si il est utilisé dans une opération de mise à jour).

5.12.2 Type UserSpecification

Description des caractéristiques (actuelles, à créer ou à modifier) d'un utilisateur.

Identifiant		Propriété	Description
name		- Attribut - Optionnel - Type : string	Attribut définissant le nom de l'utilisateur
0 à n fois	roles	- Élément - Optionnel - Type : Role	L'utilisateur peut avoir différents rôles dans l'utilisation du serveur de signature. Ces rôles lui permettent d'accéder à des opérations sur le serveur de signatures. Ces éléments sont soit de type RoleSpecification soit des RoleReference.
1 à n fois	credentials	- Élément - Optionnel - Type : Credential	Modes d'authentification qu'utilisera l'utilisateur pour s'authentifier auprès du serveur. (voir descriptif de l'élément au paragraphe 5.14).
0 à n fois	groups	- Élément - Optionnel - Type : GroupReference	Élément complexe contenant l'identification du groupe auquel l'utilisateur appartient (voir descriptif de l'élément au paragraphe 5.15.1).
0 à n fois	trustedApplication	- Élément - Optionnel - Type : Application	Élément complexe contenant l'identification des application pour lequel l'utilisateur fait confiance (donne autorisation à la délégation). (voir descriptif de l'élément au paragraphe 5.13.1).

5.13 Type « Application »

Type abstrait décrivant une Application du serveur, il se décline soit en ApplicationSpécification (description des caractéristiques d'une Application) soit en ApplicationReference (référence à une Application déjà existante sur le serveur). Une Application est aussi un User : elle peut donc être utilisée partout où le type User est utilisé (ex. élément user des GroupSpecification).

5.13.1 Type ApplicationReference

Référence à une Application existante sur le serveur.

Identifiant	Propriété	Description
applicationId	- Attribut - Optionnel - Type : ServerId	Attribut contenant l'identifiant de l'application.
value	- Élément - Optionnel - Type : ApplicationSpecification	Description des caractéristiques de l'application (ou de sa mise à jour si elle est utilisée dans une opération de mise à jour).

5.13.2 Type ApplicationSpecification

Description des caractéristiques (actuelles ou à créer ou modifier) d'une Application.

Identifiant	Propriété	Description
name	- Attribut - Optionnel - Type : string	Attribut définissant le nom de l'application
0 à n fois	roles	- Élément - Optionnel - Type : Role L'application tout comme un utilisateur peut avoir différents rôles dans l'utilisation du serveur de signature. Ces rôles lui permettent d'accéder à des opérations sur le serveur de signatures. Ces éléments sont soit de type RoleSpecification soit des RoleReference.

Identifiant		Propriété	Description
1 à n fois	credentials	- Elément - Optionnel - Type : Credential	Modes d'authentification qu'utilisera l'utilisateur pour s'authentifier auprès du serveur. (voir descriptif de l'élément au paragraphe 5.14).
0 à n fois	groups	- Elément - Optionnel - Type : GroupReference	Elément complexe contenant l'identification du groupe auquel l'utilisateur appartient (voir descriptif de l'élément au paragraphe 5.15.1).
0 à n fois	trustedApplication	- Elément - Optionnel - Type : Application	Elément complexe contenant l'identification des applications aux quelles cette application fait confiance (donne autorisation à la délégation). (voir descriptif de l'élément au paragraphe 5.13.1).

5.14 Type « Credential »

Type abstrait décrivant un mode d'authentification au serveur. Il se décline soit en CredentialSpecification (caractéristiques du mode d'authentification) soit en CredentialReference (référence à un mode d'authentification déjà créé sur le serveur).

5.14.1 Type CredentialReference

Référence à un mode d'authentification du serveur.

Identifiant	Propriété	Description
credentialId	- Elément - Obligatoire - Type : CredentialId	Identifiant du mode d'authentification. Cet identifiant est lié à l'utilisateur qui le possède et à sa nature (mot de passe certificat...)

Identifiant	Propriété	Description
value	<ul style="list-style-type: none"> - Élément - Optionnel - Type : CredentialSpecification 	Élément complexe contenant un type de credential (voir descriptif de l'élément au paragraphe 5.14.3).

5.14.2 Type CredentialId

Identifie de façon unique sur le serveur un mode d'authentification d'un utilisateur. Il est pour se faire basé sur l'identifiant de l'utilisateur qui le possède et sur une clé (unique pour utilisateur donné) calculée en fonction de la nature du mode d'authentification. Lorsqu'il est utilisé pour des opérations de création ou de mise à jour, cette clé est remplacée par la description (CredentialSpecification) que l'on souhaite créer ou modifier.

Identifiant		Propriété	Description
userId		<ul style="list-style-type: none">- Attribut- Optionnel- Type : ServerId	Identifiant de l'utilisateur du mode d'authentification.
Au Choix	credential	<ul style="list-style-type: none">- Élément- Obligatoire- Type : CredentialSpecification	Élément complexe contenant un type de credential (voir descriptif de l'élément au paragraphe 5.14.3).
	credentialKey	<ul style="list-style-type: none">- Élément- Obligatoire- Type : String	Chaine de caractères identifiant pour un utilisateur donné de façon unique un mode d'authentification existant.

5.14.3 Type CredentialSpecification

Type abstrait décrivant les caractéristiques d'un mode d'authentification. Il se décline soit en PasswordWrapper soit en X509Certificate.

5.14.3.1 Type PasswordWrapper

Description d'un mode d'authentification par mot de passe.

Identifiant	Propriété	Description
passwordHash	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : Base64Binary 	Attribut contenant la représentation hachée du mot de passe encodé en Base64
hashAlgo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : string 	Attribut contenant l'identification de l'algorithme utilisé pour hacher le mot de passe.

5.14.3.2 Type X509CertificateWrapper

Description d'un mode d'authentification par certificat X509.

Identifiant	Propriété	Description
x509Certificate	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : Base64Binary 	<p>Attribut contenant le certificat X509 encodé en Base64.</p> <p>Note : pour des données issues de fichiers PEM exclure les lignes marquant le début et la fin du Certificat.</p>

5.15 Type « Group »

Type abstrait destiné à désigner les groupes d'utilisateurs. Il se décline en GroupSpecification (description des caractéristiques d'un groupe existant à créer ou à modifier) ou GroupReference (référence à un groupe existant sur le serveur).

5.15.1 Type GroupReference

Référence à un groupe existant sur le serveur.

Identifiant	Propriété	Description
-------------	-----------	-------------

Identifiant	Propriété	Description
groupid	- Attribut - Optionnel - Type : ServerId	Attribut contenant l'identifiant du groupe.
value	- Élément - Optionnel - Type : GroupSpecification	Description du groupe voir 115

5.15.2 Type GroupSpecification

Description d'un groupe.

Identifiant	Propriété	Description
name	- Attribut - Optionnel - Type : String	Attribut contenant le nom du groupe.
groups	- Élément - Optionnel - Type : Group	Ensemble des groupes liés à celui-ci. Il peut s'agir de GroupReference ou de GroupSpecification.
users	- Élément - Optionnel - Type : User	Ensemble des utilisateurs liés à ce groupe. Il peut s'agir de UserReference, UserSpecification, ApplicationReference ou ApplicationSpecification.

5.16 Type « UsersToGroupAssociation »

Association d'utilisateurs à un groupe.

Identifiant	Propriété	Description
-------------	-----------	-------------

Identifiant		Propriété	Description
1 à N fois	userIds	<ul style="list-style-type: none"> - Élément - Obligatoire - Type :ServerId (string) 	Élément contenant l'identification d'un utilisateur
	groupID	<ul style="list-style-type: none"> - attribut - Obligatoire - Type : ServerId (string) 	Attribut contenant l'identifiant du groupe.

5.17 Type « Chain »

Ce type sert dans le cadre de l'utilisation des opérations de listage d'objets sur le serveur. Il permet de décrire les caractéristiques de recherche que l'on souhaite appliquer. La chaîne est un chaînage de contraintes liées par des opérateurs AND ou OR.

Identifiant	Propriété	Description
maxResutls	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : Int (supérieur à 0) 	Nombre maximal de réponses retournées.
constraint	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Constraint 	Première contrainte de recherche. Voir 116.
links	<ul style="list-style-type: none"> - Élément - Optionnel - Type : ChainLink 	Description du groupe voir 115

Lorsque le paramètre « maxResults » n'est pas présent, le serveur de signature retournera l'ensemble des résultats satisfaisant les contraintes (qui peut être considérable). Lorsque le nombre de réponses est limité (paramètre optionnel « maxResults » est défini), les réponses sont ordonnées en fonction de leur identifiant et seules les « maxResults » premières sont retournées. La réponse contiendra alors un indicateur « totalResutls » indiquant le nombre total de réponses présentes sur le serveur. Pour récupérer les réponses suivantes, il faudra rajouter à la chaîne une contrainte sur l'identifiant (IDRange) commençant à l'identifiant de la dernière réponse retournée.

5.17.1 Type Constraint

Description d'une contrainte à appliquer à la recherche. Il s'agit d'un type abstrait qui se décline en StringConstraint, IdRange, IdList ou BooleanConstraint.

5.17.1.1 Type BooleanConstraint

Cette contrainte s'applique à tout élément contraint de type booléen.

Identifiant	Propriété	Description
constrainedElement	- Attribut - Obligatoire - Type : String	Elément contraint.
isNegated	- Attribut - Optionnel - Type : boolean	Si activé la contrainte est négative.
value	- Attribut - Obligatoire - Type : boolean	Valeur de l'élément « true » ou « false ».

5.17.1.2 Type StringConstraint

Identifiant	Propriété	Description
constrainedElement	- Attribut - Obligatoire - Type : String	Elément contraint.

Identifiant	Propriété	Description
isNegated	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	Si activé la contrainte est négative.
type	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : StringConstraintOperator 	Type de contrainte: <ul style="list-style-type: none"> • EQUALS : valeur égale • CONTAINS : valeur contenue • BEGINS_WITH : commence par la valeur • ENDS_WITH : finit par la valeur
value	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : String 	Valeur à tester

5.17.1.3 Type IdRange

Contrainte applicable aux éléments contraints de type ServerId elle sert à définir une plage de recherche d'Id.

Identifiant	Propriété	Description
constrainedElement	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : String 	Élément contraint.
isNegated	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	Si activé la contrainte est négative.
from	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	L'id recherchée est au moins égale à cette valeur

Identifiant	Propriété	Description
to	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	L'id recherche est au plus égal à cette valeur

5.17.1.4 Type IdList

Contrainte applicable aux éléments contraints de type ServerId elle sert à définir une liste de rechercher (ou d'exclusion si isNegated est activé) d'Id.

Identifiant	Propriété	Description
constrainedElement	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : String 	Élément contraint.
isNegated	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : boolean 	Si activé la contrainte est négative.
ids	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : ServerId 	Liste des id à rechercher (ou exclure).

5.17.2 Type ChainLink

Maillon de la chaîne, association d'un opérateur de chaînage et d'une contrainte.

Identifiant	Propriété	Description
-------------	-----------	-------------

Identifiant	Propriété	Description
booleanOperator	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : BooleanOperator 	Opérateur de chainage : AND ou OR à appliquer entre le maillon précédent et ce maillon.
constraint	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Constraint 	Contrainte de recherche. Voir 116.

5.18 Type EnforcedAuthenticationMethod

Type abstrait destiné à regrouper les différentes méthodes d'authentification renforcée.

La version 1.4 ne contient qu'une seule méthode renforcée : DASAuthentication.

La version 1.4 contient deux méthodes renforcées : DASAuthentication et FIDOAuthentication.

5.18.1 Type DASAuthentication

Ce type permet de définir les paramètres nécessaires à l'utilisation de la méthode d'authentification renforcée DAS (Dynamic Activation Secret) qui est mise en œuvre avec un HSM permettant de le gérer. L'utilisateur peut choisir d'utiliser un secret d'activation dynamique basé soit sur un DAS simple ou obtenu en combinant un DAS avec le hash des données à signer. Il peut en outre fixer le nombre maximum d'utilisations de ce même secret dynamique avant expiration (dans le cas où le DAS est calculé avec le hash, sa valeur change pour chaque document différents à signer).

Identifiant	Propriété	Description
counterMax	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur supérieure ou égale à 1 	Nombre maximum d'utilisation du mot de passe dynamique avant de re-nécessiter un re-calcul du challenge.
authDataLength	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 0, 2 et 32 	Longueur des données d'authentification vérifiées.

Identifiant	Propriété	Description
authTryMax	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 1 et 255 	Nombre d'essais erronés autorisés avant blocage.
activationPeriod	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 1 et 255 	Durée maximale de validité d'une activation, en minutes.
blockingPeriod	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 1 et 1440 	Durée maximale du blocage en cas de dépassement du nombre d'erreurs, en minutes.

5.18.2 Type AuthMethodEnum

Ce type est une énumération basée sur des chaînes de caractères. Il permet de définir la méthode d'authentification à utiliser dans le cadre d'une authentification renforcée fournie avec un HSM. L'utilisation pour la signature d'une clé privée est soumise à la fourniture d'un secret partagé entre le HSM et l'utilisateur selon l'une des méthodes ci-dessous :

- une simple authentification par DAS : valeur de l'énumération 'BASIC_OTP' ;
- une authentification par combinaison d'un DAS et du hash des données à signer : valeur de l'énumération 'OTP_WITH_HASH' ;
- une authentification par token FIDO : valeur de l'énumération 'FIDO_U2F'.

5.18.3 Type FIDOAuthentication

Ce type permet de définir les paramètres nécessaires à l'utilisation de la méthode d'authentification renforcée FIDO U2F (Fast Identity Online) qui est mise en œuvre avec un HSM permettant de le gérer. L'utilisateur peut outre fixer le nombre maximum d'utilisations de ce secret dynamique avant expiration.

Identifiant	Propriété	Description
-------------	-----------	-------------

Identifiant	Propriété	Description
counterMax	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur supérieure ou égale à 1 	Nombre maximum d'utilisation du mot de passe dynamique avant de re-nécessiter un re-calcul du challenge.
authTryMax	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 1 et 255 	Nombre d'essais erronés autorisés avant blocage.
activationPeriod	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 1 et 255 	Durée maximale de validité d'une activation, en minutes.
blockingPeriod	<ul style="list-style-type: none"> - Element - Optionnel - Type : int réduit à une valeur comprise entre 1 et 1440 	Durée maximale du blocage en cas de dépassement du nombre d'erreurs, en minutes.
fidoToken	<ul style="list-style-type: none"> - Element - Obligatoire - Type : String 	Identifiant du token FIDO associé à la clé

5.19 Type ActivationData

Type complexe servant à transmettre au serveur des données d'activation de la clé. En version 1.3 il se décline en DASActivationData. En version 2.4 il se décline en FIDOActivationData.

5.19.1 Type DASActivationData

Type complexe destiné à stocker les données d'activation des clés soumises au mécanisme renforcé d'authentification DAS réalisé avec un HSM.

Identifiant	Propriété	Description
userActivationSecret	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : base64Binary 	Secret d'activation du coffre fort de signature protégeant les clés de l'utilisateur encodé en base 64.
authenticationMethod	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : AuthMethodEnum 	Choix de méthode d'authentification (voir 5.18.2).

5.19.2 Type FIDOActivationData

Type complexe destiné à stocker les données d'activation des clés soumises au mécanisme renforcé d'authentification FIDO réalisé avec un HSM.

Identifiant	Propriété	Description
userActivationSecret	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : FIDOActivationSecret 	Voir §5.11.3
keyHandle	<ul style="list-style-type: none"> - Élément - Obligatoire - Type : Base64String 	Identifiant de la clé auprès du token FIDO

5.20 Type ActivationResult

Type complexe servant à transmettre à l'appelant des données résultant de l'activation de clé. En version 1.3 il se décline en DASActivationResult. En version 2.4 il se décline en FIDOActivationResult.

5.20.1 Type DASActivationResult

Type complexe destiné à stocker les données d'activation des clés soumises au mécanisme renforcé d'authentification fournit par unHSM.

Identifiant	Propriété	Description
dynamicChallenge	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : base64Binary 	Challenge dynamique généré par le HSM encodé en Bas64.

5.20.2 Type FIDOActivationResult

Type complexe destiné à stocker les données d'activation des clés soumises au mécanisme renforcé d'authentification fournit par un HSM.

Identifiant	Propriété	Description
registeredTokens	<ul style="list-style-type: none"> - Element - Optionnel - Type : Array of RegisteredToken 	<p>Tableau des tokens enregistrés.</p> <p>Voir §5.28</p>

5.21 Type ServerId2NameInfo

Il représente l'association d'un identifiant du serveur avec un nom d'objet.

Identifiant	Propriété	Description
serverId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Identifiant serveur.

Identifiant	Propriété	Description
name	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Nom de l'objet associé.

5.22 Type KeyUsageArray

Il représente une liste d'usages de clés pour la génération de certificats.

Identifiant	Propriété	Description
digitalSignature	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : Boolean 	Booléen à true si le bit digitalSignature doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
contentCommitment	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : Boolean 	Booléen à true si le bit contentCommitment doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
keyEncipherment	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : Boolean 	Booléen à true si le bit keyEncipherment doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
dataEncipherment	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : Boolean 	Booléen à true si le bit dataEncipherment doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
keyAgreement	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : Boolean 	Booléen à true si le bit keyAgreement doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.

Identifiant	Propriété	Description
keyCertSign	- Attribut - Obligatoire - Type : Boolean	Booléen à true si le bit keyCertSign doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
crlSign	- Attribut - Obligatoire - Type : Boolean	Booléen à true si le bit crlSign doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
encipherOnly	- Attribut - Obligatoire - Type : Boolean	Booléen à true si le bit encipherOnly doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.
decipherOnly	- Attribut - Obligatoire - Type : Boolean	Booléen à true si le bit decipherOnly doit être présent, false si il ne doit pas être présent, null si on ne souhaite pas le vérifier.

5.23 Type Simples

5.23.1 ServerId

Il représente un identifiant du serveur de signature sous forme de chaîne de caractères (c.a.d. un entier supérieur à 1).

C'est donc une extension du type String limitée par l'expression régulière [1-9]\d*.

5.23.2 SignatureKeyId

Il représente l'identifiant d'une clé de signature du serveur qui identifie de manière unique chaque clé de signature d'un signataire donné. Les identifiants sont constitués de chaînes de caractère d'au moins un caractère parmi :

- les lettres non accentuées,
- les chiffres

- les signes –, +, /, et \ et _

Les espaces ne sont pas autorisés.

C'est donc une extension du type String limitée par l'expression régulière

'[a-zA-Z0-9\-\+\._\]+'

5.23.3 BooleanOperator

Il représente un opérateur de chainage booléen dans une chaîne de contraintes de recherche (voir 116). Ses valeurs possibles sont :

- AND et
- OR ou

Pour des raisons techniques, le XOR n'est plus présent depuis la version 1.3.0.

5.24 Type CertificateRequestParameters

Type abstrait destiné à regrouper les différentes méthodes de génération de certificat.

5.24.1 Type SCEPRequestParameters

Type complexe destiné à stocker les informations spécifiques liées à la requête de demande de certificat du protocole SCEP.

Identifiant	Propriété	Description
challengePassword	- Attribut - Optionnel - Type : String	La requête de certificat (PKCS#10) dans le protocole SCEP spécifie un attribut « challengePassword PKCS #9 » qui doit être envoyé dans le cadre de la demande. Cet attribut permet de faire une demande de certificat auprès du serveur SCEP (voir #GetCert dans document en Référence : draft-nourse-scep)

Identifiant	Propriété	Description
transactionID	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	L'identifiant de transaction est une chaîne générée par le client lors du démarrage d'une transaction afin d'identifier celle-ci. Cet attribut permet d'effectuer une demande de certificat qui nécessite une intervention manuelle dans l'IGC (voir #GetCertInitial dans document en Référence : draft-nourse-scep).

5.25 Type RequestCertificateResult

Type abstrait destiné à regrouper les différentes réponses des fournisseurs de certificats.

Identifiant	Propriété	Description
status	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : String 	Retourne le statut de la réponse du fournisseur de certificat.
certificate	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : base64Binary 	Le certificat de signature généré par le fournisseur.

5.26 Type SCEPRequestResult

Type complexe destiné à stocker les informations spécifiques liées à la réponse de demande de certificat du protocole SCEP.

Identifiant	Propriété	Description
transactionID	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	L'identifiant de transaction est une chaîne générée par le client lors du démarrage d'une transaction afin d'identifier celle-ci. Cet attribut permet d'effectuer une demande de certificat qui sera valorisé de manière asynchrone par l'IGC, par exemple lorsque celle-ci requière une intervention d'un opérateur de l'IGC (voir #GetCertInitial dans document en Référence : draft-nourse-scep).

Identifiant	Propriété	Description
failInfo	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : String 	Cette information est renvoyée par le serveur SCEP afin de connaître la raison de l'échec (voir #failInfo dans le document en référence : draft-nourse-scep).

5.27 Type « FIDOInfos »

Type abstrait regroupant les différentes représentations d'un token FIDO. Il se décline en FIDOInfosReference (référence à un token FIDO déjà existant sur le serveur) ou FIDOInfosSpecification (spécification d'un token FIDO non existant sur le serveur).

5.27.1 Type FIDOInfosReference

Décrit un token FIDO déjà existant sur le serveur. Il est renvoyé par les opérations de listage de token FIDO et envoyé pour les opérations de mises à jour.

Identifiant	Propriété	Description
tokenId	<ul style="list-style-type: none"> - Attribut - Optionnel - Type : ServerId 	Attribut contenant l'identifiant du token FIDO.
value	<ul style="list-style-type: none"> -Elément -Optionnel -Type : FIDOInfosSpecification 	Description des caractéristiques du token FIDO (ou de sa mise à jour si il est utilisé dans une opération de mise à jour).

5.27.2 Type FIDOInfosSpecification

Description des caractéristiques (actuelles, à créer ou à modifier) d'un token FIDO.

Identifiant	Propriété	Description
id	<ul style="list-style-type: none"> - Attribut - Obligatoire - Type : ServerId 	Identifiant du token FIDO

Identifiant	Propriété	Description
owner	- Elément - Obligatoire - Type : ServerId	Identifiant de l'utilisateur propriétaire du token
certificate	- Elément - Obligatoire - Type : Base64String	Certificat utilisé pour la génération de la signature après l'enrôlement de l'utilisateur FIDO
pubKey	- Elément - Obligatoire - Type : Base64String	Clé publique de l'utilisateur pour l'authentification FIDO
pubKey	- Elément - Obligatoire - Type : Base64String	Clé publique de l'utilisateur pour l'authentification FIDO
keyHandle	- Elément - Obligatoire - Type : Base64String	Identifiant de la clé auprès du token FIDO
challenge	- Elément - Obligatoire - Type : Base64String	Challenge associé au token FIDO

5.28 Type « RegisteredToken »

Identifiant	Propriété	Description
token	- Element - Obligatoire - Type : Base64String	Tableau contenant un token encodé en base64. [appId, challenge, version, keyHandle]

6 Critères de recherche possibles pour les différents objets

6.1 Recherches d'utilisateurs (ou d'applications)

Élément de contrainte	Clé de recherche	Type de recherches possibles
Nom	http://www.bull.security.com/Server/coreAdmin/v1.1.0/#User.name	StringConstraint
Id	http://www.bull.security.com/Server/coreAdmin/v1.1.0/#User.id	IdRange IdList

6.2 Recherches de groupes

Élément de contrainte	Clé de recherche	Type de recherches possibles
Nom	http://www.bull.security.com/Server/coreAdmin/v1.1.0/#Group.name	StringConstraint
Id	http://www.bull.security.com/Server/coreAdmin/v1.1.0/#Group.id	IdRange IdList

6.3 Recherches de politiques de signature

Élément de contrainte	Clé de recherche	Type de recherches possibles
Nom	http://www.bull.security.com/SignServer/adminSig/v1.1.2/#ServerSignaturePolicy.name	StringConstraint
Verrouillage	http://www.bull.security.com/SignServer/adminSig/v1.1.2/#ServerSignaturePolicy.locked	BooleanConstraint

6.4 Recherches de profils de signature

Élément de contrainte	Clé de recherche	Type de recherches possibles
Nom	http://www.bull.security.com/SignServer/sigServices/v1.1.0/#SignatureProfileSpecification.name	StringConstraint

6.5 Recherches de profils de génération de clé de signature

Élément de contrainte	Clé de recherche	Type de recherches possibles
Nom	http://www.bull.security.com/SignServer/adminSig/v1.1.2/#SkGenerationProfileSpecification.name	StringConstraint