siham
SYSTÈME D'INFORMATION
DES RESSOURCES HUMAINES
DANS UNE APPROCHE MUTUALISÉE

Dossier De Conception Détaillée Interfaces Ldap

Historique

Date	Version	Auteur	Statut	Commentaire(s)
19/10/2010	0.2	JDupont	Initialisation	
25/10/2010	1.0	V. Schneider	Prise en compte remarques V0.2	
08/11/2010	2.0	J. Simonnet	Prise en compte remarques V1.0	
19/11/2010	3.0	J. Simonnet	Prise en compte remarques V2.0	
25/01/2001	3.1	V. Schneider	Ajout des interfaces nomenclatures au schéma des interfaces §1.3	

Documents de référence

Type du dossier	Nom du dossier	Date
DCG	SIHAM_A01-P07_DCD_AEX_Interfaces_LDAP_V3_1.doc	29/07/2010
DCD	SIHAM_A1-P08_DCD_INT_InterfaceReferentiel.doc	xx/xx/2010

Sommaire

1.0 Introduction

1.1 Objectif et contexte du document de conception détaillée

Le but de ce document est de présenter la conception détaillée des interfaces LDAP sur le projet SIHAM.

Ce décrit, de façon détaillée, les informations nécessaires à l'implémentation des interfaces définies durant la phase de conception générale.

1.2 Principes généraux des interfaces LDAP

Dans le cadre de la mise en place des interfaces LDAP, sur le projet SIHAM, le fonctionnement général de la gestion des utilisateurs suit les étapes suivantes:

- La création d'un dossier agent dans HR Access
- La synchronisation avec le LDAP de l'établissement
- La création du compte utilisateur
- La création de l'utilisateur et son affectation au dossier agent existant dans HR Access

L'interface de création et de modification d'un utilisateur, implique ainsi l'existence d'un dossier agent crée dans SIHAM.

1.3 Interfaces LDAP – Siham par services web

Dans le cadre de l'intégration de la solution Siham à l'écosystème des établissements un mécanisme de synchronisation entre Siham et les LDAP établissement est nécessaire.

Pour des raisons d'urbanisation et de rationalisation des interfaces, cette intégration s'appuie sur les briques techniques déjà mises en œuvre dans le cadre de l'interfaçage avec le référentiel Prisme.

Le dossier de conception détaillé de l'interface référentiel (SIHAM_A1-P08_DCD_INT_InterfaceReferentiel.doc) décrit les échanges liés à la synchronisation des agents et des nomenclatures.

Les besoins identifiés sont de deux types :

- extraction SIHAM :
  - La récupération des agents modifiés,
  - La récupération des agents à échéance,
- mise à jour SIHAM :
  - La création/Modification d'un utilisateur,
  - La désactivation d'utilisateurs.

Pour satisfaire ce besoin Siham met à disposition des établissements un ensemble de Web Services à intégrer.

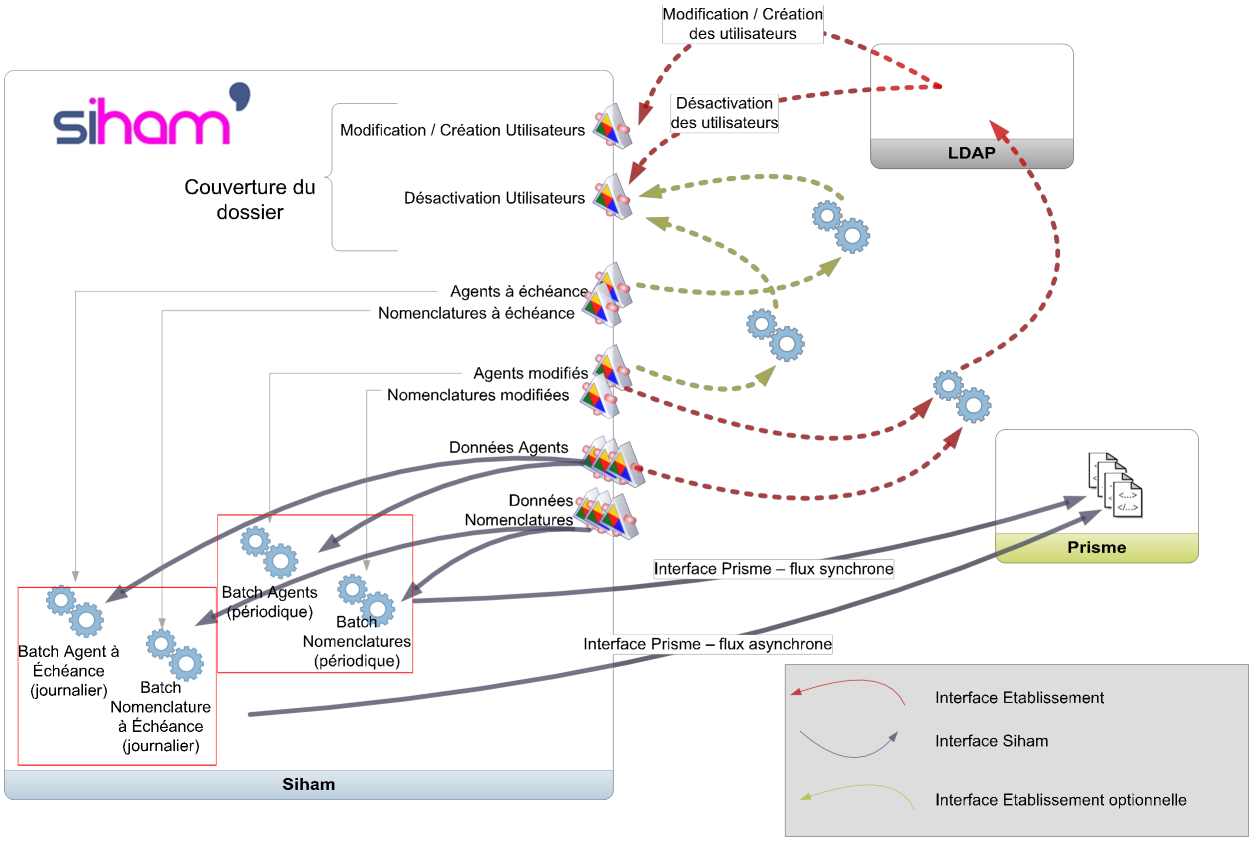


Figure 1: Synthèse des interfaces agent et utilisateurs

Ce dossier couvre les méthodes de web service de :

- Modification / création des utilisateurs,
- Désactivation utilisateur.

2.0 Le webservice d'interfaçage avec LDAP

2.1 Utilisateurs et habilitation dans HR Access

Dans le cadre de la solution Siham, où l'authentification est basée sur du SSO (LDAP, CAS ou Schiboleth) la gestion des utilisateurs et des habilitations se base sur les tables :

- ZY4I : identifiants utilisateurs,

Informations / ZY: Dossiers du personnel

< >	Nom	Libellé	Type	Uniq/Répet.	Fixe/Histo.	Modifié le	Modifié par	Verrouillé ...
	4F	Affectation expatriés/salariés étr.	Réelle	Unique	Historique	2010-05-26-13.01.38	EXPORT	
	4G	Rémun. tot. expatriés/salariés étr.	Virtuelle			2010-05-26-13.01.38	EXPORT	
	4H	Suivi des coûts d'expatriation	Réelle	Répétitive	Fixe	2010-05-26-13.01.38	EXPORT	
	4I	HR Access Identifiant Utilisateur	Réelle	Unique	Fixe	2010-05-26-13.01.38	EXPORT	
	4K	Répartition comptable	Réelle	Répétitive	Historique	2010-05-26-13.01.38	EXPORT	
	4L	Expatriation : Logement	Réelle	Répétitive	Fixe	2010-05-26-13.01.38	EXPORT	

- ZY09 pour la gestion des habilitations.

Informations / ZY: Dossiers du personnel

< >	Nom	Libellé	Type	Uniq/Répet.	Fixe/Histo.	Modifié le	Modifié par	Verrouillé ...
	00	Identification du dossier	Réelle	Unique	Fixe	2010-05-26-13.01.38	EXPORT	ASYLLA
	01	Pays-Secteur	Réelle	Unique	Historique	2010-05-26-13.01.38	EXPORT	
	05	Nom de naissance	Réelle	Unique	Fixe	2010-05-26-13.01.38	EXPORT	
	06	Prénom	Réelle	Unique	Fixe	2010-05-26-13.01.38	EXPORT	
	07	Nom usuel	Réelle	Unique	Fixe	2010-05-26-13.01.38	EXPORT	
	08	Affectations de confidentialité	Réelle	Répétitive	Historique	2010-05-26-13.01.38	EXPORT	ASYLLA
	09	Liste des Rôles	Réelle	Répétitive	Historique	2010-05-26-13.01.38	EXPORT	VSCHNEID
	0F	Adresses	Réelle	Répétitive	Historique	2010-05-26-13.01.38	EXPORT	KIQBAL
	0G	Adresse principale	Réelle	Unique	Fixe	2010-05-26-13.01.38	EXPORT	
	0H	Numéros de téléphones	Réelle	Répétitive	Historique	2010-05-26-13.01.38	EXPORT	CRAOUL

Les deux tables ZY4I et ZY09 appartiennent à la structure dossier du personnel ZY, dont la table primaire est la table ZY00. L'ensemble des autres tables de la ZY sont cascadées par rapport à cette table est liées par une clef primaire : le nudoss (numéro de dossier).

La relation entre les deux tables est une relation 1 à n : chaque utilisateur déclaré possède un ensemble d'habilitation. La clé primaire liant les deux tables est le *nudoss* (numéro de dossier).

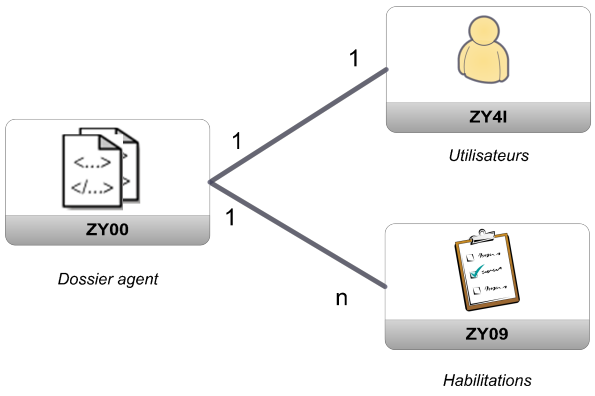
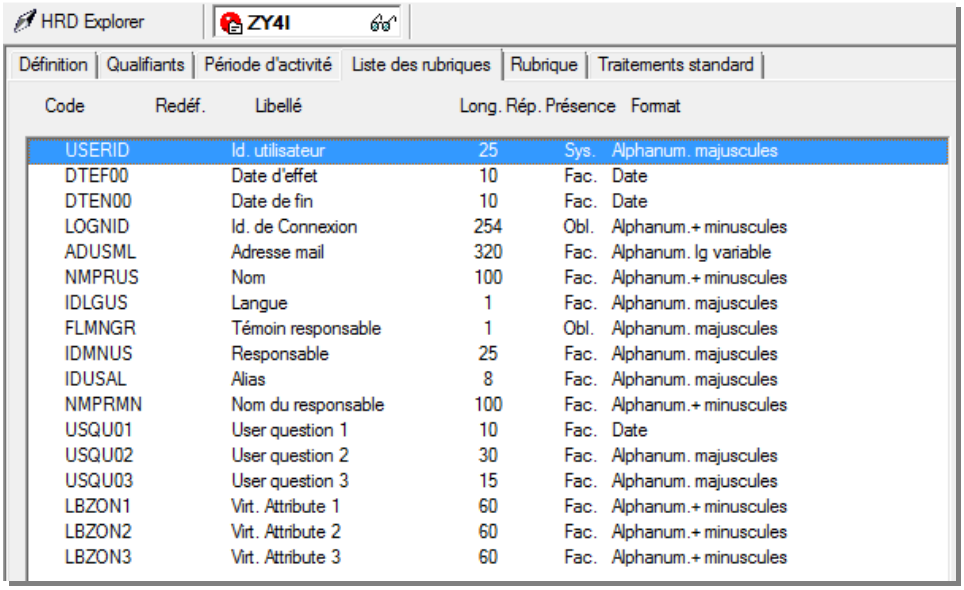


Figure 2 – Relation ZY4I / ZY09

2.1.1 La table des utilisateurs ZY4I

Le champ LOGNID correspond à l'identifiant unique poussé par la solution d'authentification.

Les champs DTEF00 et DTEN00 permettent de fixer la période de validité de l'utilisateur, en dehors de cette période le compte ne pourra se connecter à l'application.



Code	Redéf.	Libellé	Long.	Rép.	Présence	Format
USERID		Id. utilisateur	25		Sys.	Alphanum. majuscules
DTEF00		Date d'effet	10		Fac.	Date
DTEN00		Date de fin	10		Fac.	Date
LOGNID		Id. de Connexion	254		Obl.	Alphanum.+ minuscules
ADUSML		Adresse mail	320		Fac.	Alphanum. lg variable
NMPRUS		Nom	100		Fac.	Alphanum.+ minuscules
IDLGUS		Langue	1		Fac.	Alphanum. majuscules
FLMNGR		Témoin responsable	1		Obl.	Alphanum. majuscules
IDMNUS		Responsable	25		Fac.	Alphanum. majuscules
IDUSAL		Alias	8		Fac.	Alphanum. majuscules
NMPRMN		Nom du responsable	100		Fac.	Alphanum.+ minuscules
USQU01		User question 1	10		Fac.	Date
USQU02		User question 2	30		Fac.	Alphanum. majuscules
USQU03		User question 3	15		Fac.	Alphanum. majuscules
LBZON1		Virt. Attribute 1	60		Fac.	Alphanum.+ minuscules
LBZON2		Virt. Attribute 2	60		Fac.	Alphanum.+ minuscules
LBZON3		Virt. Attribute 3	60		Fac.	Alphanum.+ minuscules

Figure 3 – Champs de la table ZY4I

2.1.2 La table des habilitations ZY09

Une habilitation est caractérisée par deux éléments principaux :

- le *ROLMOD* : modèle de rôle correspondant aux droits /actions pouvant être exécutées par l'utilisateur,

Modèle de rôle	Libellé	Catégorie	Structure	Long.
ALLHRLO	Professionnel RH - Par localisa	HRREP	SLOC	2
GESTHAB	Professionnel RH - Par localisa	HRREP	SLCL	11
GESTHAB0	Gest Hab0	HRREP	SLCL	11
GESTHAB1	GEST HAB1	HRREP	SLCL	11
GESTHAB2	GESTHAB2	HRREP	SLCL	11
GESTHAB3	GESTHAB3	HRREP	SLCL	11
PADMADM	Administrateur de GP	HRREP	PADMADM	0
PCORPHR	Responsable RH groupe	HRREP	SEXP	3
PDIRCEO	Directeur général	HRREP	PDIRCEO	0
PDIRCFO	Directeur financier CFO	HRREP	PDIRCFO	0
PDIRHRD	Directeur des ressources hum	HRREP	PDIRHRD	0
PDIRLCL	Personne en charge du site	HRREP	SLCL	11
PDMSADM	Administrateur de document	HRREP	PDMSADM	0

- le *ROLVAL* : la valeur de rôle définissant le périmètre d'application du *ROLMOD*

Localisation	Libellé	
BE	Belgique	
CA	Canada	
CH	Suisse	
DE	Allemagne	
FE	Fonction publique	
FR	France secteur pri	
FT	Fonction publique	
IN	International	
IT	Italie	
MA	Maroc	
NL	Pays Bas	
PT	Portugal	
SP	Espagne	
UK	Royaume uni	
US	Etats Unis	

Les champs *DTEF00* (date d'effet) et *DTEN00* (date de fin) permettent de fixer la période de validité de l'habilitation, en dehors de cette période l'utilisateur n'est pas habilité pour le rôle donné.

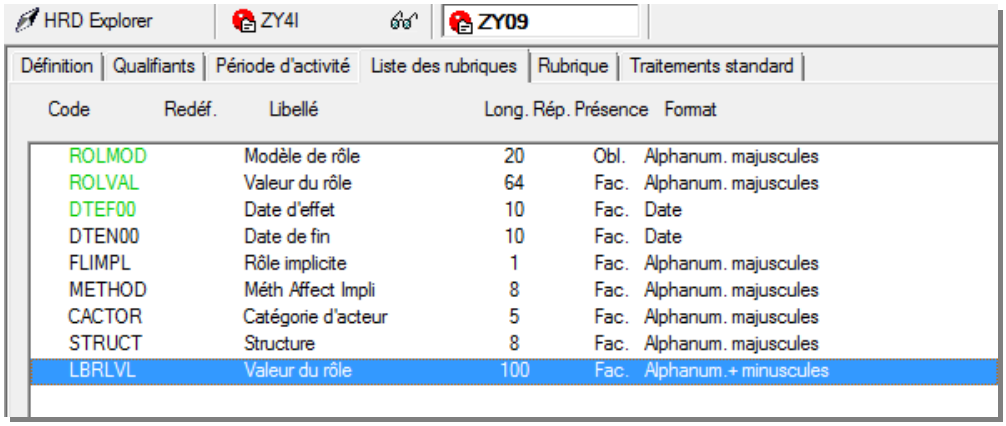


Figure 4 – Champs de la table ZY09

2.1.3 Rôle métier et accès self-service

Les droits d'accès au self-service ne sont pas gérés dans la table *ZY09* qui ne contient que les habilitations métiers. L'accès aux informations personnelles est directement hérité de l'habilitation de l'utilisateur :

- si l'utilisateur a une période valide dans la *ZY4I* il peut se connecter et a accès au self service,
- si l'utilisateur n'a pas une période valide il ne peut se connecter à l'application et n'a donc pas accès au self service.

Un utilisateur n'ayant aucune habilitation métiers en n'étant pas autorisé à accéder son dossier personnel ne peut se connecter à l'application, car il n'a plus de droit.

2.2 Opération 1 : Création/Modification d'un utilisateur

2.2.1 Rappel

Ce web service permet la création/modification d'un utilisateur dans Siham et sont affectation à un dossier agent existant dans SIHAM.

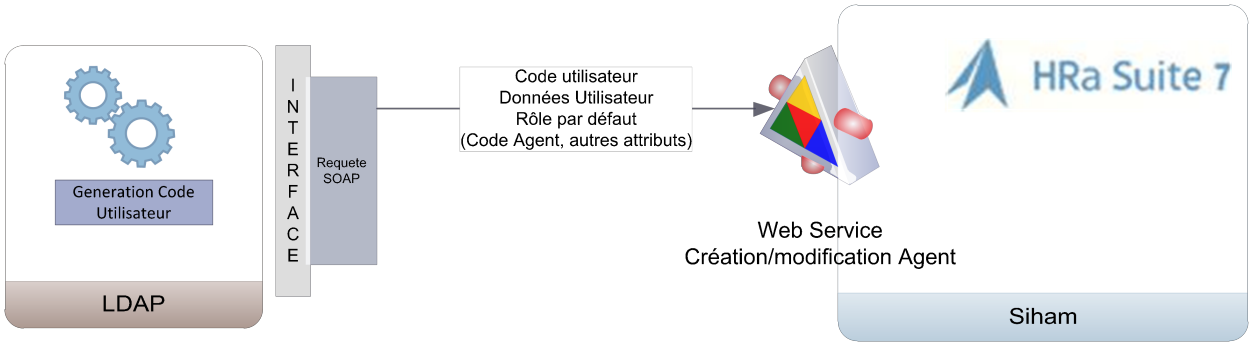


Figure 5 - Web Service - Création/Modification d'un utilisateur

2.2.2 Spécification du Webservice

S'il existe une entrée correspondante au matricule (dossier agent) dans la ZY00, la méthode récupère le nudoss correspondant.

- Si une occurrence du nudoss existe dans la ZY4I
  - et que le LOGNID est cohérent avec celui passé en paramètre, la méthode applique la modification demandée dans la ZY4I :
    - adresse mail et nom usuel si spécifié,
    - DTEF00=hier et DTEN00=hier si l'option d'activation du self-service est à false,
    - DTEF00=aujourd'hui et DTEN00=31/12/9999 si l'option d'activation du self-service est à true.
  - si le LOGNID ne correspond pas à celui placé en paramètre, la méthode génère un exception.

- Si une aucune occurrence du nudoss n'existe dans la ZY4I, la méthode la créée avec les paramètres soumis par l'interface : login, adresse mail et nom usuel. Elle positionne les dates d'effet et fin comme suit :
  - DTEF00=hier et DTEN00=hier si l'option d'activation du self-service est à false,
  - DTEF00=aujourd'hui et DTEN00=31/12/9999 si l'option d'activation du self-service est à true.

S'il n'existe pas d'entrée dans la ZY00 pour l'agent spécifié, la méthode renvoie une exception.

2.2.2.1 Données en entrée de l'opération

Nom de la donnée	Description	Nom technique	Type	Obligatoire
lognid	Code utilisateur	lognid	String	Oui
matcle	Code Agent (matricule)	matcle	String	Oui
selfsrv	WS Option de SS	selfsrv	Boolean	Oui
adusml	Adresse email	adusml	String	Non
nmprus	Nom usuel	nmprus	String	Non

2.2.2.2 Données en sortie de l'opération

En cas de bon fonctionnement de l'interface la méthode renvoie le matcle.

Nom de la donnée	Description	Nom technique	Type	Obligatoire
matcle	Code Agent (matricule)	matcle	String	Oui

Si en incident se produit elle renvoie une exception.

2.3 Opération 2 : Désactivation Utilisateur

2.3.1 Rappel

Ce Web Service modifie les habilitations pour un Code Agent / Codes Utilisateurs donnés:

- soit en lui supprimant toutes ses habilitations,
- soit en lui supprimant toutes ses habilitations sauf son accès au Self Service,
- l'option est fixée comme paramètre de fonctionnement de l'interface.

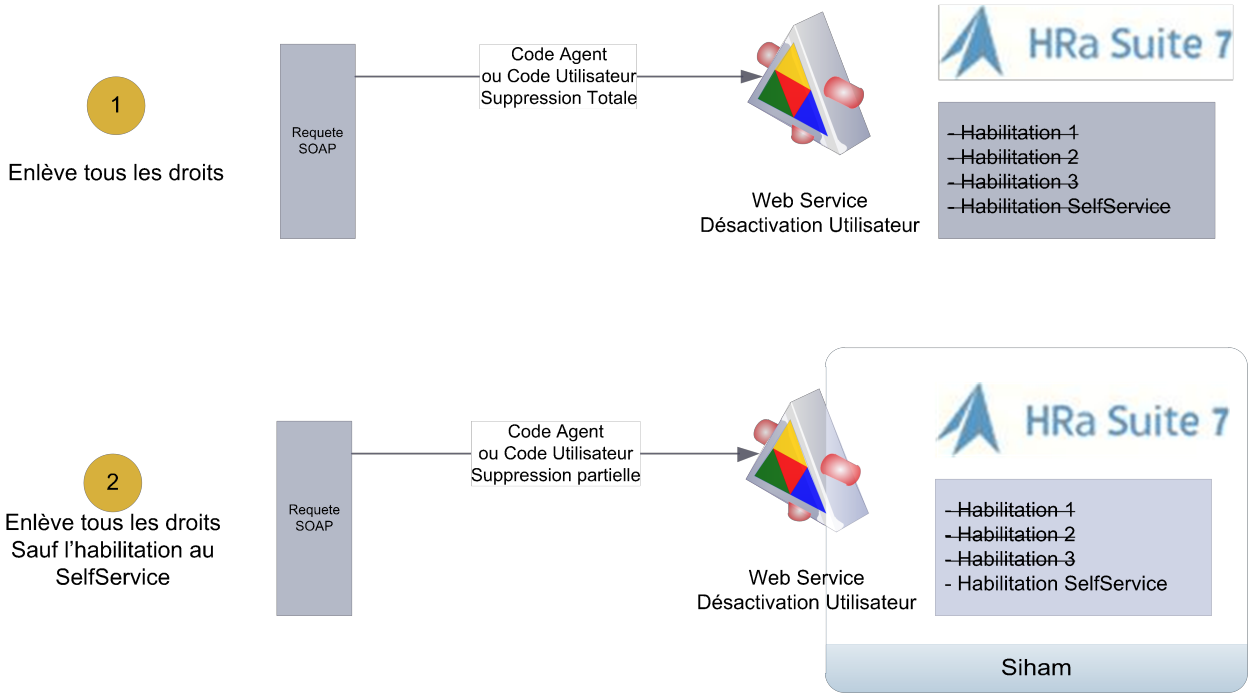


Figure 6 - Web Service - Désactivation Utilisateur

2.3.2 Détails de l'opération

Si le « *typecle* » transmis est «*LOGNID*», la méthode déduit que le deuxième argument de la méthode correspond au *lognid* de l'utilisateur. Elle recherche alors dans la table *ZY4I* le *nudoss* de l'occurrence correspondant au *lognid*.

- Si aucune occurrence n'est trouvée, la méthode renvoie une exception,
- Si plusieurs occurrences sont trouvées, la méthode renvoie une exception,
- Si un seul nudoss correspond, la méthode modifie la date *DTEN00* de l'ensemble des habilitations de la *ZY09* de même nudoss, pour les mettre à la date de la veille du jour de l'exécution du traitement. Si de plus le paramètre *selfsrv* de la méthode est false, la date *DTEN00* de l'occurrence de la *ZY4I* est mise à la date de la veille du jour de l'exécution du traitement.

Si le « *typecle* » transmis est «*MATCLE*», la méthode déduit que le deuxième argument de la méthode correspond au *matcle* de l'utilisateur. Elle recherche alors dans la table *ZY00* le *nudoss* de l'occurrence correspondant au *matcle*.

- Si aucune occurrence n'est trouvée, la méthode renvoie une exception,
- Si plusieurs occurrences sont trouvées, la méthode renvoie une exception,
- Si un seul nudoss correspond, la méthode modifie la date *DTEN00* de l'ensemble des habilitations de la *ZY09* de même nudoss, pour les mettre à la date de la veille du jour de l'exécution du traitement. Si de plus le paramètre *selfsrv* de la méthode est false, la date *DTEN00* de l'occurrence

de la *ZY4I* est mise à la date de la veille du jour de l'exécution du traitement.

2.3.3 Spécification du Webservice

2.3.3.1 Données en entrée de l'opération

Nom de la donnée	Description	Nom technique	Type	Obligatoire
typecle	Type de clef transmise l'utilisateur : « MATCLE » ou « LOGNID »	typecle	String	Oui
cleuser	Identifiant utilisateur (matricule ou login)	cleuser	String	Oui
selfsrv	WS Option de SS	selfsrv	Boolean	Oui

2.3.3.1 Données en sortie de l'opération

Nom de la donnée	Description	Nom technique	Type	Obligatoire
lognid	Code utilisateur	lognid	String	Oui

2.4 Sécurité des web services

2.4.1 Principe de l'implémentation de la sécurité des web services

La sécurisation des web services s'appui sur les besoins de :

- authenticité du service,
- authentification des services accédants,
- confidentialité des flux échangés

La mise en œuvre de cette sécurisation est possible grâce à :

- Implémentation du SSL,
- Intégration web des services au reverse proxy Apache,
- Implémentation Spring Security et d'une authentification login/mot de passe

L'implémentation de la sécurité est faite sur les 3 couches en répondant aux 5 principes généraux de sécurité:

	Confidentialité	Intégrité	Identification	Authentification	Autorisation
Couche réseau			Filtrage IP		

Couche Transport	SSL/TLS(*)	SSL/TLS(*)	SSL/TLS(*)		
Messagerie SOAP			Spring Security	Spring Security	Spring Security (fichiers plats)

Figure 7 - Web Service – Sécurisation des Web Services

Les échanges par Web Services en zone sécurisée et se connectant directement au serveur applicatif Tomcat n'ont pas besoin de connexion SSL, cela permet de plus de garantir les performances des échanges.

2.4.2 Principes de Spring Security

Spring Security permet de gérer l'accès aux ressources d'une application Java. Ces ressources peuvent être des pages web, mais aussi des objets de services métier.

Toute ressource sollicitée par un appelant est rendue accessible si, d'une part, l'appelant s'est identifié, et si d'autre part, il possède les droits nécessaires (des rôles dans le vocabulaire Spring Security).

Les requêtes HTTP sont interceptées par un **filtre de servlet** qui délègue à un bean Spring les traitements de vérification d'accès aux pages web.

Ce bean met en oeuvre une chaîne de filtres. Chacun des filtres est un bean auquel est attribuée une tâche précise :

- Intégration dans la session HTTP des informations de sécurité contenues dans la requête
- Vérification de l'identité de l'appelant et affichage d'une invite de connexion si nécessaire
- Vérification des droits d'accès à la ressource sollicitée

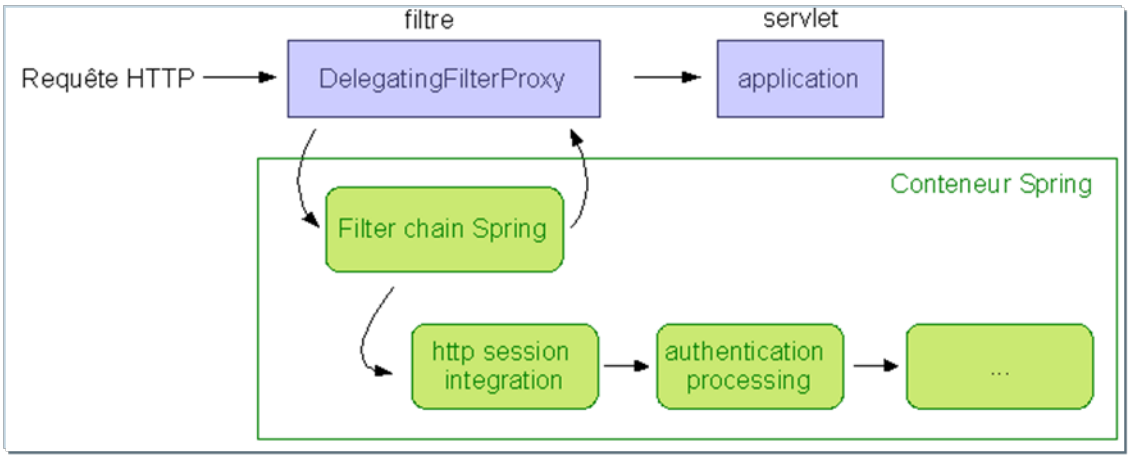


Figure 8 - Web Service – Spring Security

2.4.1 Implémentation de Web Services avec l'API OpenHR

L'API OpenHR, est une API Java livrée avec HR Access qui permet de créer des applications tierces appelées "applications clientes OpenHR" (ou applications clientes), pour une utilisation en mode TP (temps réel)

OpenHR offre différents services dont le principal est de permettre l'accès en lecture/écriture et en temps réel aux données gérées par le serveur HR Access tout en réutilisant la logique fonctionnelle existante de contrôle des données (implémentée sous la forme de programmes COBOL). OpenHR fournit un modèle objet intuitif et cohérent avec les concepts HR Access (dossier, information, occurrence, rubrique, etc.).

L'API OpenHR communique avec un serveur nommé serveur OpenHR. Cette communication s'effectue de manière synchrone à l'aide de sockets. Le protocole de communication entre client (l'application bâtie au-dessus de l'API OpenHR) et serveur OpenHR est propriétaire de HR Access : il est constitué d'un ensemble de messages au format texte. Le serveur OpenHR agit comme une porte d'entrée vers le monde HR Access : toute communication de l'API avec le serveur HR Access passe obligatoirement par lui. Le serveur OpenHR est une application Java stand-alone, c'est-à-dire non hébergée dans un conteneur JEE, chargé de démarrer et d'arrêter les programmes COBOL du serveur HR Access.

L'API OpenHR agit donc comme un middleware pour se connecter au serveur HR Access.

Dans le cadre de l'implémentation de Web Services du projet SIHAM, les composants techniques sont les suivants:

- La compilation des sources et la gestion des dépendances sont assurées par **MAVEN2**
- **Spring**, le framework JAVA open source est utilisé par HRAccess (Conteneur léger, AOP, injection de dépendance, inversion de contrôle…)
- **Spring Security** permet la sécurisation du Web Service: toutes les requêtes sur le WebService demandent une authentification de type BASIC pour aboutir

- **Apache CXF 2.2** comme moteur SOAP
- **L'API OPENHR** pour la communication avec HRAccess

3.0 Annexes

3.1 Annexe 1 : Template de contrat de service

SIHAM_A01-P07_DC
D_AEX_Interfaces_LI