
Installation du package esup-portail

Version uPortal-2.5.x-esup

Ce document décrit l'installation et le paramétrage du package du socle 'esup-portail' issu de la version 2.5 d'uPortal, à **partir de la version 2.5-esup-2**.

Le package contient l'installation d'un serveur Tomcat (facultatif), de la distribution uPortal 2.5 et des bibliothèques et personnalisations propres à l'environnement esup-portail (voir le document sur les [groupes esup-portail](#)).

Il permet de simplifier considérablement le paramétrage de base d'uPortal.

Le package ne livre pas les différents canaux esup-portail, à installer indépendamment.

Vincent MATHIEU
[Université Nancy 2](#)

Dates de modification

Revision version 1.0	17 décembre 2005	
Revision version 1.01	12 janvier 2006	Ajout de la directive esup.monitor
Revision version 1.0.2	9 février 2006	Ajout d'exemples dans les tâches ant
Revision version 1.0.3	7 mars 2006	Modifications liées au changement de packaging
Revision version 1.0.4	2 juin 2006	le keystore par défaut est maintenant esup-portail.keystore
Revision version 1.0.5	22 octobre 2006	ajout de la propriété esup.render.wai
Revision 1.0.6	5 mars 2007	Rectification du lien vers l'utilitaire 'groupload'
Revision 1.0.7	7 mars 2007	Changement lien de téléchargement (sourcesup)

-
- [1. Liens utiles](#)
 - [2. Conseils et remarques préalables](#)
 - [3. Pré-requis](#)
 - [3.1. Système d'exploitation](#)
 - [3.2. SGBD](#)
 - [3.3. Environnement JAVA](#)
 - [3.4. ANT](#)
 - [3.5. Authentification](#)
 - [3.6. Annuaire LDAP](#)
 - [4. Contenu du package](#)
 - [5. Organisation du package](#)
 - [5.1. Organisation File système suite au décompactage du package](#)
 - [5.2. Autres répertoires créés par le package](#)
 - [6. Installation et personnalisation](#)
 - [7. Lancement / arrêt du serveur](#)
 - [7.1. Lancement 'non root'](#)
 - [7.2. Comptes 'locaux'](#)
 - [8. Propriétés du fichier esup.properties](#)
 - [8.1. Paramètres de chemins file système](#)
 - [8.2. Paramètres http et ports TCP liés à esup-portail](#)
 - [8.3. Paramètres liés au serveur Tomcat](#)
 - [8.4. Paramètres liés au host virtuel \(url publique\)](#)
 - [8.5. Paramètres liés au host réel](#)
 - [8.6. Paramètres liés à LDAP](#)
 - [8.7. Paramètres liés à CAS](#)
 - [8.8. Paramètres liés aux accès SGBD uPortal](#)
 - [8.9. Paramètres divers](#)
 - [9. Directives ant](#)
 - [9.1. Directives développées par esup-portail](#)
 - [9.2. Directives liées à la distribution uportal](#)
 - [10. Utilisation de l'arborescence 'Custom'](#)
 - [11. Conseils d'utilisation](#)
 - [11.1. Mots de passe initiaux](#)
 - [11.2. Configuration Tomcat](#)
 - [11.3. Applications admin et manager de tomcat](#)
 - [11.4. Personnalisations](#)
-

1. Liens utiles

- + [Documentation de l'ancien package 2.5](#)
- + [Changements par rapport au package 2.5 antérieur à la version 2.5-esup-2](#)
- + [Chargement du package](#)
- + [Installation depuis le repository subversion](#)
- + [Changelog](#)
- + [conseils d'installation](#)
- + [procédure de migration d'une version uPortal-2.4.x-esup vers uPortal-2.5.x-esup](#)

2. Conseils et remarques préalables

L'utilisation de l'installation d'Esup-portail issue de ce package nécessite une bonne connaissance préalable d'uPortal, de l'environnement Esup-portail et de l'authentification CAS.

Si ce n'est pas le cas, la bonne stratégie de prise en main est d'utiliser le [package de test et développement d'esup-portail](#) (sous windows ou linux), en suivant les étapes suivantes :

- + Dans un premier temps, utiliser sans l'authentification CAS (`esup.cas.auth=false` et `esup.cas.proxy=false`), et avec l'environnement uPortal (`esup.env.esup-portail=false`). On se retrouve alors dans un environnement proche de la distribution quick-install d'uPortal, avec certains apports esup (paramétrages LDAP, ...).

Ceci permet d'appréhender le fonctionnement du portail, dans un environnement permissif (droits élevés aux comptes créés), et avec des canaux de tests. En particulier, il est important de comprendre le fonctionnement du gestionnaire de groupes uPortal.

- + Ensuite, y ajouter l'authentification CAS (`esup.cas.auth=true` et `esup.cas.proxy=true`) ; s'assurer que le portail fonctionne correctement en mode proxy.
- + Enfin, paramétrer l'environnement esup-portail (`esup.env.esup-portail=true`, et `ant esup.db.init`); on se retrouve alors dans un environnement très proche de celui proposé par ce package.

Lorsque vous serez familiarisé avec ce dernier environnement, vous pourrez appréhender le package esup-portail sans problème.

3. Pré-requis

L'installation de ce package suppose les pré-requis suivants :

3.1. Système d'exploitation

Linux, UNIX

En pratique, esup-portail a été testé sur Redhat xx et solaris xxxx

3.2. SGBD

Fonctionne avec un SGBD acceptant du SQL standard et les transactions, et proposant un drivers JDBC natif.

Testé avec mysql (version 4 ou supérieure), postgree, oracle.

Note pour mysql

ce sgbd ne supporte des transactions qu'avec des tables de type *BDB* ou *INNODB* ; il est donc impératif d'utiliser ce type de tables pour la base esup-portail.

3.3. Environnement JAVA

Un *JDK 1.5* doit être préalablement installé sur le (ou les) serveur supportant le portail.

La variable **JAVA_HOME** doit être évaluée, et le chemin `$JAVA_HOME/bin` rajouté au **PATH**

3.4. ANT

Le logiciel *ANT* doit être installé, avec une version égale ou supérieure à 1.6.5.

La variable **ANT_HOME** doit être évaluée, et le chemin `$ANT_HOME/bin` rajouté au **PATH**

3.5. Authentification

Un serveur **CAS** doit être opérationnel au sein de l'établissement.

3.6. Annuaire LDAP

L'établissement doit disposer d'un annuaire **LDAP** exhaustif et compatible supann.

4. Contenu du package

Ce package permet une installation *uPortal* en y ajoutant des librairies et un environnement spécifiques esup-portail.

Il apporte également les librairies et le paramétrage permettant l'utilisation du mécanisme de *SSO CAS*.

Il installe et paramètre un serveur Tomcat ; cette installation reste optionnelle (mais conseillée), afin de permettre à des établissements désirant utiliser un autre moteur de servlet de le faire.

5. Organisation du package

Ce paragraphe décrit l'organisation File Système liée au package. Il précise l'organisation juste après le décompactage du package, puis celle qui résulte de la commande `ant esup.unzip`, qui installe réellement le package.

5.1. Organisation File système suite au décompactage du package

Le décompactage du package crée le répertoire `uPortal-2.5.x-esup-2-y` qui est la racine de l'installation (`esup.root`).

Sous cette racine, après exécution de la commande `ant esup.unzip`, on trouve les fichiers et répertoires suivants :

esup.properties

c'est le fichier principal permettant le paramétrage de l'installation. Il est à personnaliser.

build.xml

c'est le fichier qui contient la logique des différentes directives des tâches ANT du package. **Ne pas modifier.**

env.sh

c'est un fichier script shell appelé par les autres scripts fournis par le package. Il est mis à jour automatiquement lors de la procédure d'installation en fonction des informations du fichier `esup.properties`. **Ne pas modifier.**

start-esup.sh, stop-esup.sh

Ce sont les scripts de lancement / arrêt du serveur esup-portail ; ils sont à adapter localement.

esup-portail.keystore

c'est un magasin de certificats qui contient le certificat de l'autorité racine du CRU et le certificat de l'AC Cybertrust ; il est utile aux établissements de l'enseignement supérieur.

scripts

C'est un répertoire qui contient des scripts utilitaires (traitements statistiques, ...)

packages

C'est un répertoire, qui contient les 'sous-packages' en format binaire qui sont installés lors du `ant esup.unzip` : le serveur tomcat, la distribution uPortal et les personnalisations esup-portail.

UpdateEsup

C'est un répertoire, qui contient les fichiers du sous-package de personnalisation esup-portail.

C'est l'ajout esup-portail par rapport aux distributions uPortal et Tomcat.

Il est essentiellement composé de 3 sous répertoires, Tomcat, uPortal, ROOT. Ces sous-répertoires contiennent des fichiers (fichiers de propriétés, librairies, sources, ...) qui seront déployés vers la distribution uPortal (`uportal.distrib`) lors du `ant esup.init`, ou vers l'environnement tomcat (`server.root`) et la racine de l'environnement exécutable (`esup.deploy`) lors du `ant esup.deploy`. **Ne pas modifier le contenu de ce répertoire.**

Custom

C'est un répertoire, qui est également composé des 3 sous répertoires Tomcat, uPortal et ROOT.

Ces répertoires sont vides lors de l'installation du package.

Ils sont déployés comme les sous-répertoire d'`UpdateEsup` (en fait, l'emplacement de ce répertoire est paramétrable : propriété `esup.custom`).

L'intérêt est de permettre aux établissements d'y mettre leurs propres fichiers de configuration, sources, ... sans intervenir directement sur les fichiers livrés avec le package.

5.2. Autres répertoires créés par le package

Ce sont des répertoires qui vont être créés et peuplés lors des directives `ant esup.init` ou `ant esup.deploy`.

{server.home} : Déploiement de tomcat

C'est le répertoire dans lequel la procédure d'installation va déployer le serveur tomcat. La procédure d'installation va également supprimer les applications d'exemples livrés avec la distribution tomcat.

Les fichiers contenus dans les répertoires `UpdateEsup/Tomcat` et `Custom/Tomcat` viendront écraser les fichiers de la distribution Tomcat.

Remarque : le déploiement de ce répertoire (et les différents paramètres liés) ne sera effectué que si le paramètre `server.deploy=true`

{esup.distrib} : Déploiement de l'environnement source d'uPortal

C'est le répertoire dans lequel la procédure d'installation va déployer les sources d'uPortal, puis ultérieurement les personnalisations liées à esup-portail et aux directives du fichier `esup.properties`.

Les fichiers contenus dans les répertoires `UpdateEsup/uPortal` et `Custom/uPortal` viendront écraser les fichiers de la distribution uPortal.

{esup.deploy} : Déploiement des exécutables esup-portail

C'est le répertoire où seront écrits les fichiers issus du déploiement esup-portail suite à leur compilation, lors de la commande `ant uportal.deploy`. C'est donc ce répertoire qui sera connu du moteur de servlet (`tomcat`) pour l'exécution du portail.

Remarque : Le déploiement d'esup-portail se fera dans un sous-répertoire nommé uPortal, celui des éventuels portlets

dans d'autres sous-répertoires dédiés.

6. Installation et personnalisation

Ce paragraphe décrit de manière rapide le processus d'installation, de paramétrage ; les paragraphes suivants détaillent les différents paramètres ou options.

L'installation complète peut (et devrait) se faire depuis un compte utilisateur non root. Il est bien sûr nécessaire que ce compte ait un droit d'écriture dans les différents répertoires utilisés.

Décompresser l'archive

```
tar -xvzf uPortal-2.5.x-esup-y.tar.gz
```

. x correspond à la sous-version d'uPortal dont est dérivé le package, y la version esup de ce package

un répertoire est créé : `uPortal-2.5.x-esup-y` ; on appellera ce répertoire la 'racine' du package (`{esup.root}`).

Remarque : il est possible d'installer depuis le dépôt subversion ; voir le [document explicatif](#).

Personnalisations principales

```
cd uPortal2.5-esup-y
```

. Adapter le fichier `esup.properties`. C'est le principal travail à effectuer. Ce fichier est décrit plus loin dans ce document.

Décompresser les sous-packages

```
ant esup.unzip
```

. Ceci décompacte les 3 sous-packages inclus dans le répertoire `packages` :

- Le serveur Tomcat, vers le répertoire `{server.home}`, si `server.deploy = true`
- Les sources uPortal, vers le répertoire `{esup.distrib}`
- Les bibliothèques, sources et personnalisations esup, vers le répertoire `{esup.root}/UpdateEsup`.

Appliquer les personnalisations

```
ant esup.init
```

Recopie les bibliothèques et personnalisations esup du répertoire `UpdateEsup`, puis celles du répertoire `Custom` vers l'installation source uPortal, puis exécute les paramètres du fichier `esup.properties` pour mettre à jour les différents fichiers de paramétrage uPortal.

Compilation des sources uPortal et esup-portail

```
ant uportal.compile
```

Cette tâche ant compile les sources java si nécessaire (si la date des sources est plus récente que celle des classes compilées).

Cette tâche n'est pas obligatoire, car les tâches suivantes en dépendent. La compilation des sources est donc effectuée si nécessaire lors des phases suivantes.

Test de l'accès à la base esup-portail

```
ant uportal.dbtest
```

Permet de tester et de valider l'accès à la base de données uPortal.

- Si le sgbd est *mysql* ou *postgree*, noter le "Database version" (dans les première lignes du rapport), et mettre cette valeur complète dans la propriété `esup.db.db-version` du fichier `esup.properties`.

Si un autre sgbd est utilisé, il faudra éditer manuellement le fichier `dbloader.xml`

Initialisation de la base esup-portail

```
ant esup.db.init
```

Initialise la base esup-portail, et déploie les canaux et fragments de base.

Déploiement vers l'environnement de production

```
ant esup.deploy
```

Déploie les différents fichiers nécessaires vers l'environnement de production du serveur J2EE.

Déploie et modifie également les fichiers du répertoire Custom/Tomcat vers le répertoire d'installation de Tomcat, si la propriété `server.deploy` a la valeur `true`.

Remarque : Un seul compte local est réellement utilisable au démarrage : le compte `admin` (mot de passe `admin`). C'est le compte de l'administrateur d'esup-portail.

7. Lancement / arrêt du serveur

Deux scripts shell sont fournis, installés sous la racine du package `{esup.root}` :

start-esup.sh

Lancement du serveur tomcat. En production, il faudra adapter ce script afin d'optimiser le fonctionnement et de fournir à la JVM les paramètres adaptés à l'environnement local.

stop-esup.sh

Arrêt du serveur tomcat.

7.1. Lancement 'non root'

Pour des raisons de sécurité, le serveur J2EE peut (et devrait) être lancé depuis un compte non root ; pour cela, il est nécessaire que les ports TCP ouverts par ce moteur (`{server.port.shutdown}`, `{server.port.http}`, `{server.port.jk}`) soient supérieurs à 1024.

Ceci ne pose pas de problème dans une configuration où un serveur apache est utilisé en frontal de ce serveur, via APJ13, ce qui est fortement conseillé.

7.2. Comptes 'locaux'

Par défaut, le serveur esup-portail authentifie CAS et avec sa base locale. Le compte d'administration est `admin`. Il est possible de se connecter avec ce compte, à l'aide d'un formulaire qui n'est pas visible depuis la page du portail.

Cette page est accessible depuis un sous-répertoire `private` de l'URL du portail, donc à l'URL `#{esup.host.https}/private/`

Par exemple, si le portail est accessible à `http://ent.univ.fr`, la page permettant de se connecter avec la base locale sera accessible à `http(s)://ent.univ.fr/private`.

Pour des raisons de sécurité, il est conseillé en production de désactiver les authentifications liées à la base locale : `esup.db.auth=false`.

8. Propriétés du fichier esup.properties

Un fichier situé à la racine du package permet de paramétrer l'ensemble de l'installation esup-portail : c'est `esup.properties`. La prise en compte de modifications dans ce fichier nécessite un `ant esup.init`, puis `ant esup.deploy`, et enfin un redémarrage du serveur.

Ces propriétés supportent toutes des valeurs par défaut, sauf la première.

8.1. Paramètres de chemins file système

java_home

C'est le répertoire d'installation de la JVM.

esup.root

C'est le répertoire où le package a été décompacté.

Par exemple, si le package a été décompacté dans `/install/esup`, `esup.root` aura pour valeur `/install/esup/uPortal-2.5.x-esup-y`

Par défaut : `/home/esup/uPortal-${esup.distrib.version}`

esup.distrib

c'est l'emplacement source et de compilation de la distribution esup-portail.

Par défaut : `/home/esup/uPortal`

esup.deploy

C'est le répertoire de déploiement d'exploitation de l'environnement esup-portail et des portlets éventuelles. Le portail sera déployé dans un sous-répertoire `uPortal` de `esup.deploy`, les portlets dans des sous-répertoires dédiés.

Par défaut : `/home/esup/webapps`

esup.custom

C'est le répertoire contenant les personnalisations propres au site.

Par défaut : `${esup.root}/Custom`

server.home

le répertoire d'installation du serveur tomcat livré avec ce package, si `server.deploy=true`

server.temp

c'est un répertoire temporaire du serveur, utilisé par exemple lors des uploads de fichiers.

Par défaut : `${server.home}/temp`

esup.log.file

nom du fichier de log uportal.

Par défaut : `${esup.root}/portal.log`

esup.stats.file

nom du fichier de stats.

Par défaut : `${esup.root}/stats.log`

esup.keystore

C'est le magasin de certificats chargé par la JVM qui supporte Tomcat, pour valider les accès https générés par les applicatifs J2EE. Il doit au moins contenir un certificat validant le serveur CAS.

Par défaut : `${esup.root}/ac-racine-cru.keystore`

A noter : un keystore est fourni avec ce package : `ac-racine-cru.keystore` ; il contient le certificat de l'ac-racine du CRU. Si vos certificats de serveurs sont délivrés par le CRU, en particulier le certificat du serveur CAS, vous pouvez utiliser ce fichier.

8.2. Paramètres http et ports TCP liés à esup-portail

Certaines des propriétés proposées ici peuvent paraître redondantes ; il n'en est rien, certaines ne sont utiles que dans le cas d'un fonctionnement en répartition de charge.

Afin de mieux comprendre le rôle de ces propriétés, voici résumées les différentes mises en oeuvre possibles du portail :

- + Seul tomcat est utilisé pour délivrer le contenu ; il doit donc implémenter le connecteur http tomcat. Configuration déconseillée en production.
- + Un serveur apache est installé en frontal du serveur tomcat ; `apj13 (mod_jk)` est utilisé pour la communication frontal - tomcat
- + répartition de charge (load-balancing), avec différentes possibilités :
 - . un seul frontal apache qui gère le load-balancing avec `mod_jk`
 - . Un load-balancer qui distribue les requêtes http vers les serveurs tomcat
 - . idem précédent, mais avec un apache en frontal de chaque tomcat

Pour ce dernier cas, on distingue le host et l'url 'virtuels' ou publiques, qui correspondent à l'url connue du public, du host et url 'réel' qui correspondent au serveur réel qui délivre le contenu.

En particulier, les requêtes https (par exemple, le retour du PGT du serveur CAS) s'effectuent vers le host réel.

8.3. Paramètres liés au serveur Tomcat

tomcat.http

C'est une propriété booléenne. Si `true`, le connecteur http de tomcat est actif. Ce connecteur n'a pas besoin d'être actif si un frontal apache avec `mod_jk` est utilisé

Défaut : `true`

tomcat.jk

C'est une propriété booléenne. Si `true`, le connecteur APJ13 de tomcat est actif. A activer uniquement si un frontal apache est utilisé.

Défaut : `true`

tomcat.port.shutdown

c'est le port de 'shutdown' de tomcat. A modifier par exemple si plusieurs instances tomcat tournent sur la même machine.

Défaut : `8005`

tomcat.port.http

c'est le port d'écoute du connecteur http de tomcat. Si le fonctionnement est de type 'standalone', donc sans frontal apache, ca sera le port d'accès au serveur esup-portail

Défaut : `8080`

tomcat.port.jk

c'est le port d'écoute du connecteur APJ13 de tomcat. Il sera utilisé par un frontal HTTP (apache, par exemple, avec `mod_jk`)

Défaut : `8009`

8.4. Paramètres liés au host virtuel (url publique)

Ces paramètres permettent de reconstruire l'URL complète d'accès au portail, connue du public.

esup.public.host

c'est le nom DNS d'accès au portail. Ce paramètre est à changer obligatoirement

Défaut : `ent.univ.fr`

esup.public.proto

le protocole utilisé pour les accès publics. Deux valeurs possibles : `http` ou `https`.

Défaut : `http`

esup.public.port

le port TCP pour les accès public au portail. valeurs possibles : "" ou ":nnn". ex : ":8080", ou ":8443".

Si valeur vide, sous-entend ":80" en http, ou ":443" en https (si l'accès 'normal' au portail est en https)

Défaut : ""

esup.public.uri

l'uri d'accès au portail. Si le portail est accessible à la racine du site, ne pas mettre "/" ; laisser vide.

Défaut : `/uPortal`

Note

L'url publique d'accès au portail se déduit de ces 4 paramètres : `${esup.public.proto}://${esup.public.host}${esup.public.port}${esup.public.uri}`

8.5. Paramètres liés au host réel

Ils sont utilisés lors d'accès https directs vers le serveur réel, soit pour l'URL de callback CAS, soit pour le passage temporaire en https.

esup.real.host

le nom DNS du serveur réel. Ne pas décommenter si pas d'utilisation en load-balancing.

Défaut : `${esup.public.host}`

esup.real.port.https

Le port TCP pour les accès https directs au serveur réel valeurs possibles : "" ou ":nnn".

ex : ":8443". Si valeur vide, sous-entend ":443".

Défaut : ""

esup.real.uri

l'uri d'accès au portail, par le serveur réel en https. Mêmes regles que `esup.public.uri`

Défaut : `${esup.public.uri}`

Note

L'url d'accès https au serveur réel se déduit de ces 3 paramètres : `https://${esup.real.host}${esup.real.port.https}${esup.real.uri}`

8.6. Paramètres liés à LDAP

Ce sont des paramètres liés à l'authentification LDAP (ne devrait pas être utile, avec CAS), et à l'accès aux informations LDAP pour constituer des groupes ou des attributs uportal issus de requêtes LDAP.

esup.ldap.host, esup.ldap.port, esup.ldap.baseDN

ce sont les informations nécessaires à la connection LDAP

Défaut : ldap.univ.fr, 389, "ou=People,dc=univ,dc=fr"

esup.ldap.bindDN, esup.ldap.bindPasswd

ces propriétés sont obligatoires, mais peuvent être laissées vides. Elles sont utilisées si besoin d'un bind non anonyme lors de la récupération des attributs LDAP liés à la personne.

Défaut : "" et ""

esup.ldap.groups.etu.formation

c'est le nom de l'attribut LDAP qui permet de 'discriminer' les formations d'étudiants

Défaut : etud-formation

8.7. Paramètres liés à CAS

Ce sont des paramètres liés à l'authentification CAS.

esup.cas.host

C'est le nom DNS du serveur CAS.

Défaut : auth.univ.fr

esup.cas.port.https

C'est le port d'accès TCP (nécessairement https) au serveur CAS.

valeurs possibles : "" ou ":nnn". ex : ":8443". si valeur vide, sous-entend ":443"

Défaut : ""

esup.cas.uri

l'uri d'accès au serveur CAS ex : esup.cas.uri=/cas.

Si le serveur CAS est à la racine http, ne pas mettre '/' : laisser vide

Défaut : ""

8.8. Paramètres liés aux accès SGBD uPortal

esup.db.auth

Valeur booléenne. Si true, on utilise également les comptes internes à la base uPortal (admin, par exemple). Ceci n'est pas indispensable.

Défaut : true

esup.db.persondirs.use

Valeur booléenne. Utilise également la base interne uPortal pour les informations d'attributs de personne.

Défaut : false

esup.db.username

c'est un nom d'utilisateur connu du SGBD, ayant tous les droits dans la base uPortal (décrite dans l'URL de connexion).

Défaut : sa

esup.db.password

c'est le mot de passe de l'utilisateur précédent.

Défaut : ""

esup.db.jdbcDriverJar

c'est le fichier .jar driver jdbc du SGBD utilisé. Ce driver doit se trouver dans le répertoire UpdateEsup/drivers (ou Custom/driver). Pour le moment, les drivers disponibles sont relatifs aux SGBD mysql, postgree, le driver oracle n'étant pas distribuable en libre.

Défaut : mysql-connector-java-3.0.15-ga-bin.jar

esup.db.className

C'est la classe java correspondant au driver jdbc.

Défaut : com.mysql.jdbc.Driver

esup.db.url

C'est l'url jdbc de connexion.

Défaut : jdbc:mysql://mysql.univ.fr/uportal

esup.db.db-version

c'est la version du serveur SGBD utilisé. Indifférent pour hsql. A remplir obligatoirement pour mysql ou postgree. Récupérer la valeur Database version lors du `ant uportal.dbtest`

Défaut : 4.1.9-max-log

8.9. Paramètres divers

esup.multiservers

Valeur booléenne. Si true, le fonctionnement multi-serveurs est activé. Nécessaire dans un fonctionnement en load-balancing.

Défaut : false

esup.webservices.axis

Valeur booléenne. Si true, les librairies axis permettant les webservices sont activées.

esup.webservices.axis.groups

Valeur booléenne. Si true, le webservice permettant d'exporter les groupes uPortal est activé.

Défaut : false

esup.monitor

Valeur booléenne. Si true, active la servlet EsupMonitor.

Cette servlet est nécessaire au fonctionnement du canal "Informations ENT".

Elle expose des informations de fonctionnement de l'ent, et éventuellement des utilisateurs connectés à l'ENT.

L'accès à cette servlet devrait être contrôlé.

Défaut : false

esup.host.logicalName

C'est un nom logique de l'instance d'uPortal. Utilisé par exemple dans les logs pour les discriminer en fonction de l'instance.

Défaut : esup1

esup.render.columns

Valeur booléenne. Si true, la navigation onglets - colonnes est celle par défaut dans uPortal. Sinon, c'est un système de menus / sous-menus alternatif.

Voir le [document lié à cette option](#).

Voie également la propriété esup.render.wai

Défaut : true

esup.render.wai

Valeur booléenne. si true, met en place les styles pour que le portail devienne conforme WAI.

Si true, force la propriété esup.render.column à false

Défaut : true

esup.session.lifetime

Duree de la session esup-portail, en minutes

Défaut : 30

esup.languages

les langues supportées par le portail. ex : en_US,ja_JP,sv_SE,de_DE

Défaut : fr_FR,en_US

esup.pubchan

Valeu booléenne. Si *true*, les canaux et les fragments fournis avec la distribution sont automatiquement publiés, lors du `ant esup.db.init`

Défaut : true

server.deploy

Valeur booléenne. Indique si le serveur Tomcat doit être installé et paramétré.

Défaut : true

esup.title.main

le 'title' html de la page esup-portail (en utf8 si accentués)

Défaut : esup-portail

esup.users.autocreate

Valeur booléenne. si true, esup-portail accepte automatiquement tout utilisateur qui a été authentifié.

Défaut : true

esup.userprefs.save

Valeur booléenne. Indique si les preferences utilisateurs doivent être sauvegardees a la fin de chaque session.

Défaut : true

esup.log.size

la taille du fichier log, avant rotation.

Défaut : 50000KB

esup.log.level

le niveau de log (INFO, WARNING, DEBUG, ...)

Défaut : INFO

esup.stats.fnames

contient la liste des 'fname' des canaux pour lesquels on désire un suivi statistique. C'est une liste de 'fname' séparés par le caractère 'virgule' (,).

Par exemple : `esup.stats.fnames=esup-webdav,pers-esup-annu,esup-mailto.`

La valeur `all` active les statistiques de service pour tous les canaux.

Par défaut : `all`

esup.stats.attributeType

c'est le nom de l'attribut uportal utilisé pour discriminer les utilisateurs dans les informations statistiques.

Défaut : `eduPersonPrimaryAffiliation`

esup.stats.size

la taille du fichier stats, avant rotation

Défaut : 30000KB

En résumé, les paramètres à priori systématiquement à modifier sont :

`java_home`, `esup.root`, `esup.host.http`, `esup.public.host`, `esup.db.username`, `esup.db.password`, `esup.db.url`, `esup.ldap.host`, `esup.ldap.port`, `esup.ldap.baseDN`, `esup.ldap.groups.etu.formation`

Et `esup.db.db-version` probablement nécessaire si `mysql` et `postgree`.

Important

Remarque sur la directive `server.deploy`

Il est préconisé de passer cette directive à la valeur `false` après mise en production. Ceci permet de passer ensuite la commande `ant esup.deploy` alors que le serveur est actif.

Cela évite également la suppression du serveur Tomcat lors d'un `ant esup.cleanall`

9. Directives ant

Les différentes actions liées à ce package (en dehors du lancement ou de l'arrêt du serveur) se font par l'intermédiaire de l'utilitaire `ant`. L'exécution de ces directives doit se faire nécessairement depuis la racine du package (`esup.root`).

Certaines directives sont directement issues de la distribution uPortal ; elles sont préfixées par `'uportal'`.

Les autres ont été créées spécifiquement pour le package `esup-portal` ; elles sont préfixées par `'esup'`. Elles exploitent les paramètres écrits dans le fichier `esup.properties`.

9.1. Directives développées par esup-portal

ant esup.unzip

Comme indiqué précédemment, cette directive décompacte les 'sous-packages' situés dans le répertoire `packages`. A noter que Tomcat n'est déployé que si `server.deploy=true`.

ant esup.init

Cette directive applique la personnalisation `esup-portal` au package :

+ nettoyage éventuelle d'un `ant esup.init` précédent.

- + copie de `updateEsup/uPortal` vers l'environnement de la distribution uPortal
- + Mêmes opérations pour `Custom/Tomcat` et `Custom/uPortal`
- + Application des paramètres du fichier `esup.properties` vers l'environnement source uPortal

Cette directive doit être suivie d'un `ant esup.deploy` et d'une relance du serveur pour prise en compte.

ant esup.cleanall

Cette directive supprime les installations ou modifications faites par l'installation précédente du package sauf le répertoire d'exécution du portail (`esup.deploy`) et Le répertoire d'installation Tomcat, si `server.deploy=false`.

ant esup.db.init

Cette directive écrase et réinitialise la base uPortal.

Si `esup.pubchan = true` (valeur par défaut), des canaux et des fragments 'de base' sont également publiés ; ce sont les canaux décrits dans les fichiers xml présents dans `properties/chanpub`, et le fichier de fragment `properties/all/esup-fragments.xml`.

ant esup.deploy

compilation si nécessaire des sources 'fraîches', et copie dans l'environnement d'exécution tomcat : `${esup.deploy}/uPortal`

Recopie de `custom/Tomcat`, puis application des paramètres du fichier `esup.properties` vers l'environnement Tomcat si `server.deploy=true`).

ant esup.users.ldapadd

Cette directive permet de 'précharger' les comptes utilisateurs uPortal à l'aide de comptes issus d'un annuaire LDAP. Elle doit être lancée avec les paramètres nécessaires aux accès LDAP :

- + `LdapURL` : obligatoire . l'URL d'accès au serveur LDAP
- + `LdapFilter` : obligatoire. le filtre LDAP à appliquer
- + `LdapLoginID` : facultatif. Un Distinguish Name (DN) en cas de bind non anonyme
- + `LdapPass` : facultatif. Le mot de passe en cas de bind non anonyme

Exemples :

```
esup.users.ldapadd -DLdapURL="ldap://ldap.univ.fr/dc=univ,dc=fr"
-DLdapFilter="(&(objectclass=inetorgperson)(cn=a*))"

esup.users.ldapadd -DLdapURL="ldap://ldap.univ.fr:392/dc=univ,dc=fr"
-DLdapFilter="(&(objectclass=inetorgperson)(cn=a*))"
-DLdapLoginID=uid=admin,dc=univ,dc=fr -DLdapPass=password
```

Il est possible de relancer régulièrement cette commande afin de créer les comptes uPortal correspondant aux nouveaux comptes LDAP ; les anciens comptes ne sont pas impactés.

Cette commande est utile en cas d'utilisation de l'option `esup.users.autocreate=false`, ce qui permet de limiter la population ayant accès au portail.

ant esup.groups.load

Cette directive permet de gérer les groupes uportal à l'aide d'un fichier de configuration xml. Il supporte le paramètre `dataFile`, qui permet d'utiliser un autre fichier de configuration.

Par défaut, le fichier utilisé est `properties/groups/GroupLoad.xml` Voir la [documentation de cet utilitaire](#).

Exemple :

```
ant esup.groupload -DdataFile=/properties/groups/esupGroupLoad.xml
```

ant esup.groups.perm.load

Cette directive permet de gérer les permissions liées aux groupes uportal à l'aide d'un fichier de configuration xml.

Par défaut, le fichier utilisé est `properties/groups/GroupPermLoad.xml`

Il supporte le paramètre `dataFile`, qui permet d'utiliser un autre fichier de configuration. Exemple :

```
ant esup.groups.perm.load -DdataFile=/properties/groups/GroupPermLoadSample.xml
```

9.2. Directives liées à la distribution uportal

Ce sont les directives qu'on peut lancer habituellement depuis la distribution uPortal. Le principal intérêt de les reprendre est d'éviter d'avoir à se déplacer dans le répertoire de distribution uportal. Le nom de ces directives est le même que les directives fournies avec la distribution uPortal, en les préfixant de 'uportal.'.

Ainsi, les directives suivantes sont fournies :

```
uportal.projecthelp,    uportal.all,    uportal.clean,    uportal.compile,    uportal.deploy,    uportal.dist,
uportal.initportal,    uportal.db,    uportal.dbtest,    uportal.pubchan,    uportal.pushfragment,
uportal.deployPortletApp, uportal.dbunload, uportal.javadoc, uportal.md5passwd, uportal.deluser.
```

Remarque : ne plus utiliser la directive `uportal.initportal`. C'est la directive `esup.db.init` qui doit être utilisée pour initialiser la base.

En pratique, les directives uPortal dont vous aurez besoin sont :

ant uportal.compile

compilation des sources, et recopie dans `${esup.distrib}/build`

ant uportal.deploy

compilation si nécessaire des sources 'fraîches', et recopie dans l'environnement d'exécution tomcat : `${esup.deploy}/uPortal`

Remarque : ne pas utiliser ; utiliser plutôt `esup.deploy`.

ant uportal.dbtest

test de la connectivité à la base

ant uportal.pubchan

permet de publier d'exécuter un/des fichiers de publication de canaux se trouvant sous `${esup.distrib}/properties/chanpub/`

exemple, pour le fichier `properties/chanpub/myChannel.xml` :

```
ant uportal.pubchan -Dchannel=myChannel.xml
```

ant uportal.pubfragments

permet de publier un fichier de publication de fragments se trouvant sous `${esup.distrib}properties/al/`

exemple, pour le fichier `properties/al/esup-fragments.xml`

```
ant uportal.pubfragments -DfragmentFile=/properties/al/esup-fragments.xml
```

ant uportal.md5passwd

permet de créer un utilisateur uPortal *local*, ou de modifier son mot de passe.

exemple, pour créer l'utilisateur local toto, ou pour modifier son mot de passe :

```
ant md5passwd -Dusername=toto
```

ant uportal.deluser

permet de supprimer un utilisateur uportal.

exemple, pour supprimer l'utilisateur toto :

```
ant uportal.deluser -Duser=ashenoy
```

10. Utilisation de l'arborescence 'Custom'

Le but est de permettre de modifier des fichiers de configuration, de source, ... tout en laissant la possibilité de faire ensuite une mise à jour du package, ou une nouvelle installation en récupérant facilement les modifications locales.

L'emplacement de ce répertoire est paramétrable (propriété `esup.custom`).

L'arborescence Custom est découpée en 3 sous-répertoires, Tomcat, uPortal et ROOT, vides initialement. Elle est donc organisée comme le répertoire UpdateEsup.

Lors d'un `ant esup.init`, les arborescences `UpdateEsup/Tomcat`, `UpdateEsup/uPortal` sont copiés dans l'environnement source uPortal ; ensuite, l'arborescence `custom` est copié vers l'environnement source uPortal ; enfin, les personnalisations sont appliquées dans les fichiers de configuration.

Lors d'un `ant esup.deploy`, l'arborescence `UpdateEsup/ROOT` est copiée à la racine de l'environnement d'exécution du portail.

Les fichiers de personnalisations qui seraient propres à l'établissement (sources java, fichiers de config, fichiers Tomcat, logos, ...), sont à placer dans l'arborescence `custom`.

Ceci vous permet de faire les modifications des fichiers de propriétés non prévues dans le package, et de faire des mises à jour ultérieures de la distribution esup-portail, sans perdre vos personnalisations.

11. Conseils d'utilisation

11.1. Mots de passe initiaux

Il est **impératif** de changer le mot de passe de l'administrateur local (`admin`) très rapidement. Et faire de même pour le compte `demo`, qui est un 'modèle' pour les comptes qui seront créés dynamiquement.

11.2. Configuration Tomcat

Il est plus que probable que chaque établissement aura à personnaliser son environnement tomcat, et sera sensible à ce que l'installation tomcat se soit pas perturbée par des modifications de paramètres liés à esup-portail.

Il est donc fortement conseillé de mettre la propriété `server.deploy` à la valeur `false` à la suite d'une première installation.

11.3. Applications admin et manager de tomcat

Lors de l'installation du package, ces applications livrées avec tomcat sont actives. Elles permettent d'administrer le serveur tomcat depuis un navigateur ; C'est un problème de sécurité potentiel.

Il est à la charge de l'établissement de les sécuriser, voire de les supprimer.

Modifier en particulier le fichier `tomcat-users.xml` afin de changer les comptes ayant des accès à ces applications.

11.4. Personnalisations

Ce package est destiné à une première installation d'un environnement de production esup-portail. Les paramètres prévus dans le fichier `esup.properties`, et les différents fichiers proposés par esup-portail seront sans doute insuffisants pour une personnalisation plus poussée.

Par exemple, lors de l'adaptation / création d'un nouveau skin, de logos, voire de modifications de sources uPortal ou esup-portail,...

Il est donc important de mettre les fichiers 'locaux' dans l'arborescence Custom : ceci permet d'une part, de bien discerner les personnalisations locales par rapport à l'environnement initial, et d'autre part, de pouvoir faire des mises à jour du package tout en pouvant ré-appliquer ces personnalisations ultérieurement.