Installation - esup-webdav-srv

ESUP WebDAV server

Installation
Auteur : (http://)

- ESUP WebDAV server
- Introduction
- Important notes
- Components
- Pre-requisite
- Naming conventions
- Installation
  - Getting the package
  - Preparing the deployment
  - Deploying the server
  - Customizing the deployment
  - Starting and stopping the server
  - Re-configuring the server
  - The ANT targets
  - Logging
    - WebDAV events logging
    - Other logging
  - Recommended configuration
    - Apache/Tomcat
    - JVM parameters
  - In case of problems...
  - Links in this page

Introduction

About the V5.2 version

The ESUP consortium has released a V5.2 version of the ESUP WebDAV server with a new store using Shibboleth, a new uPortal store which use a new webservice and a filter which allow to redirect to another url.

The purpose of this document is to explain how to install the ESUP WebDAV server. Many features have been modified or improved since the previous versions. You will find a FAQ (under construction) at the end of this document. Please _DO NOT contact directly the developpers_ neither by mail nor by phone. Use instead the injac-utilisateurs@esup-portail.org mailing list. When you request the mailing list please ensure to attach the logs of the server in debug mode AND your build.properties file (remove every critical entry). See the section "Logging".

Important notes

The packaging and configuration file have been slightly modified since the previous version. Read carefully this documention before deploying the server.

Components

The ESUP WebDAV server is composed of : * a [Jakarta Slide|http://jakarta.apache.org/slide/] V2.1

- a [Jakarta Tomcat|http://tomcat.apache.org/] 5.5.25 server
- an uPortal group store
- a shib store
- a quota listener
- a quota management Web service
- an authentication layer to manage SSO-CAS, LDAP and Trusted authentication
- an administration tool: ESUPWDServerManager

Pre-requisite

- an ESUP-Portail 2.5 or higher version - with the "esup-portal-ws" Web service deployed and running (see [ja-sig wiki abour esup-portal-ws wiki|http://www.ja-sig.org/wiki/display/UPC/esup-portal-ws+-+a+web+service+to+publish+groups+and+user+attributes])
- an LDAP directory (compliant with the Inet Org Person standard (SUPANN for France))
- a jdk 1.5 - you can encounter problems trying to deploy the server with a jdk 1.4. Indeed, starting from the 5.2 version we use librairies compiled with a jdk1.5

Naming conventions

- package.home : directory where the package is unzipped.
- deploy.home : directory where the server is deployed.

Installation

Getting the package

Download the package esup-serveur-WebDav-xxx in the download page. Unzip it in a choosen directory.

Preparing the deployment

Open and edit the package.home/esup-serveur-WebDav-xxx/build.properties file

> It is STRONGLY recommended to fill this file in carefully given that some parameters (written as FINAL) can NOT be re-configured once the server is deployed and started.
>
> Each parameter HAS TO BE filled in except those written as OPTIONAL
>
> You can keep the value of parameters written as DEFAULT=defaultValue - the others MUST be modified

build.properties files :

The build.properties file contains all of the explanation needed to fill it in. This document focuses only on some points.

Tomcat/Apache parameters * webdavServer.url : URL used by WebDAV clients such as the administration tool - change this value according to your system configuration (especially if you have an apache running in front of the WebDAV server)

LDAP parameters * slide.ldap.Scope : this parameter has change since the last version - scope = SCOPE_SUBTREE_LEVEL | SCOPE_ONE_LEVEL | SCOPE_OBJECT_LEVEL

- slide.ldap.Filter : replace & by & and | by |

backend parameters * slide.content****Store : path where to store Slide content

- slide.metadata****Store : path where to store Slide metadata

> It is strongly recommended to store Slide content and metadata on a fast machine.

> If you want to keep data of a previous installed version of the server (higher than V3), keep the same values for these parameters.

ESUP-Portail portal connection * slide.uPStore.DisableCache : you are advised against disabling the cache, this could affect both the WebDAV server and the portal performance

root user * slide.rootUser : administrator uid (must be registered in the LDAP)

authentication router propertiesPlease fill in this section carefully. Most of the problems are due to a wrong configuration of the authentication layer. security

> The security of the ESUP WebDAV server is performed by both an IP adresses control and a password control. The authentication layer ensures that the client trying to connect to the server is one of the portal IP adresses and that the password contained in the request is correct.

- slide.trustedFilter.password : enter here your trusted password - this password is used by the storage channel and administration tool (ESUPWDServerManager) to access the ESUP WebDAV server. It must be complex enough.
- slide.trustedFilter.users : leave this parameter empty
- axis.trustedFilter.password : you have to set the same password as slide.trustedFilter.password
- axis.trustedFilter.users : you can restrict here the people who can access the quota Web service and then use the ESUPWDServerManager administration tool (user1:user2:user3:...)

Deploying the server

Execute the ant unzip, ant init and ant deploy targets, in this order.

Customizing the deployment

You can personalize your installation thanks to the perso directory ("ESUP-like" packaging). You will find in particular log4j.xml files to configure logging (see the logging section for more details).

Starting and stopping the server

Execute the server.start and server.stop ANT targets.

Re-configuring the server

Modify the package.home/esup-serveur-WebDav-xxx/build.properties files and run the ANT init and ANT deploy targets. Restart the server.

The ANT targets

The ANT targets are very close to the ones present in the ESUP-Portail portal.

Run the ANT help target to have a short description of the most common used targets.

- *buildEclipseProject* : for developpers, check out the project from the SN (trunk), run the buildEclipseProject and unzip targets and you will have a well configured environment to modify the source code.
- *buildZip* / *buildTar* : build packaging of the server (with Slide and Tomcat - no Internet connection is required to install the server).
- *clean* : for developpers, clean the project
- *compile* : for developpers, compile the application, administrator do not need to run this target given that it is called by the deploy target.
- *config* : quick reconfigure the server, but use ANT init and ANT deploy instead.
- *deploy* : deploy the server - for a full deployment run ANT unzip, ANT init and ANT deploy.
- *getComponents* / *removeComponents* : retrieve remote files and libraries, administrator do not need to run this target given that it is called by the init target.
- *help* : display a quick help
- *init* : initialize the application for compilation
- *runClient* : do nothing... 🙂
- *unzip* : unzip Tomcat in the deployment directory and Slide in the working directory
- *server.start* / *server.stop* : start and stop the server

Logging

WebDAV events logging

WebDAV events logging are configured by the slide.webdavLogging.* parameters in the package.home/esup-serveur-WebDav-xxx/build.properties file.

Other logging

More generally you can configure logging using log4j.xml files in the package.home/esup-serveur-WebDav-xxx/perso directory (do not forget to init and restart the server). By default, this directories contains files to log the authentication layer, uPortal store (group management), quota listener (quota management) and the administration tool. You can add entries as needed.

Recommended configuration

It seems (we're not sure of that, it's a recommendation) to be more convenient for some webdav clients to use "" as the context path of slide. That is to say, http://webdav.univ-rennes1.fr for example should be directly the webdav server (and not http://applis.univ-rennes1.f/mywebdav for example). It can avoid some authentications problems for some OS (Windows, MacOs or others) For that, in build.properties put slide.context = Be carreful to not put a "/", just nothing. The goal is to make a 'Context path="" ' in the server.xml file for slide application.

You can use Slide with or without a web server behind Tomcat. There are many reasons to use a web server behind Tomcat : * You have already others Applications on your server.

- It can be also for deploying load balancing.
- It's a good way also to make HTTPS (SSL).
- It's very useful to make many virtual hosts and configure some authentication possibilities (http://webdav-ldap.univ-rennes1.fr, http://webdav-noauth.univ-rennes1.fr, http://webdav-trusted.univ-rennes1.fr, ...), see comments in build.properties in the section about web.xml and the authentication filters.
- Others ...
  But just keep in your mind that you can use Tomcat Standalone (without another webserver in frontend).

Apache/Tomcat

Before to configure an Apache/Tomcat binding make sure that you have no problems using webdav directly on Tomcat. Indeed, you can use Slide in 2 modes together (in the same time), that is to say with and without web server frontend. Look at build.properties for options.

The "web server" is usually Apache. You can use different version of Apache : * Apache 1.3,

- Apache 2.0,
- Apache 2.2.

And you can use also different ajp13 connector. See official documentation for that : * mod jk 2 -> http://tomcat.apache.org/connectors-doc-archive/jk2/index.html

- mod jk -> http://tomcat.apache.org/connectors-doc/
- AJP Proxy (> Apache 2.1) -> http://tomcat.apache.org/connectors-doc/proxy.html

If using Apache2.2, use together AJP Proxy. Apache2.2 is a very new version of Apache, so it's not included in the stable packages suite of distributions like debian for example 😞

mod jk 2 is no more supported, use mod jk instead

Apache2 is easier to configure, if you can, use it.

When you use a frontend web server behind Tomcat, HTTP requests and responses are alterate (keep in your mind, some options can be important).

With mod jk 2, mod jk and (not sure, AJP Proxy seems to be better) AJP Proxy, you can have authentication problem between apache and some mricrosoft webdav client (mostly Windows XP). Indeed, the session doesn't persist between 2 requests, so authentication prompt is always displayed (even if authentication works well). You can notice this in debug mode (put org.esupportail.filter in debug mode via log4j.xml), AuthenticationRouter class logs the session id for each request.

Tips (maybe it can help, but we're not sure at all :S) : * If you use load balancing : *** Modify ther server.xml file in adding jvmRoute="jvm1" to the Engine tag :
<Engine name="Catalina" defaultHost="localhost" debug="0" jvmRoute="jvm1"></Engine>

-
  -
    - Next you should define the jvm1 jvm route in your mod jk, mod jk2 or ajp proxy configuration. For exemple for mod jk, you will have a workers.properties with this line :
      worker.ajp13.jvm_route=jvm1 (you can use another name that ajp13 for worker, see official doc).

- On the client try to connect with a dummy user, next try to connect with a real user. You can try to cancel first and next to try to login really ...
- Maybe the most useful Tip here: Use HTTPS instead HTTP, that is to say configure your Apache Server to make SSL (via the mod ssl). It seems there is no more problem of lost sessions (tested with Apache2.0 and mod jk). Note however that some webdav clients don't support HTTPS.

JVM parameters

The parameter -Ddrools.compiler=JANINO must be add to the JVM in order to use the store shib.

In case of problems...

Read and read again the documentation and comments in the build.properties file. Most of the problems are due to a wrong configuration.
Read the FAQ.
Log the server in debug mode.
Contact the injac-utilisateurs mailing list providing your build.properties file (remove critical entries) and your logs in debug mode.
Pray... 😉

Links in this page

- [the
  ESUP download page|http://www.esup-portail.org/ProjetsAboutis/]