CAS clearPass

Rappel - ProxyCAS

Le ProxyCas classique nécessite que le service cible soit cassifié.

Pour ce faire, on peut cassifier l'authentification du service cible (serveur sftp, serveur imap , ...) via pam_cas par exemple -ou encore mod_cas d'apache pour un service web livré par apache, etc..

Dans ce cadre, uPortal, qui agit en proxyCAS, peut alors transmettre à la portlet qui le demande (telle qu'email-preview, esup-filemanager, ...) un casProxyTicket qui permettra une authentification sur le serveur CAS.

Ce mécanisme (et la mise en place) est décrit sur cette page ci-avant.

Limitations du ProxyCAS

D'autres services proposent une authentification moins souple, non maîtrisée, non cassifiable.

C'est le cas notamment des services proposés par les solutions Microsoft Windows: Microsoft Exchange, montages CIFS Windows.
Ceux-ci ne sont pas cassifiables et donc le proxyCas ne peut fonctionner avec eux.

Solutions abordables éventuellement envisageables (!)

Une solution abordable (hors mécanismes kerberos et similaires) est alors de revenir à une authentification classique : s'authentifier sur ces services avec le username/password de l'utilisateur. Il faut donc que la portlet en ait connaissance.

On peut alors :

- redemander à l'utilisateur de ressaisir son password dans un formulaire dédié à la portlet ;
- utiliser un password commun/générique (voire un username/password commun/générique)
- récupérer le password en clair depuis une base de données, le ldap, ...
- ... ou encore récupérer le password en clair depuis le serveur CAS

Cette dernière option s'appelle le **clearPass CAS**.

⚠️ Suivant vos contraintes en matière de SSI, ces différentes solutions ne sont pas forcément toutes acceptables.

Mise en place du clearPass CAS

Le clearPass CAS permet à uPortal de récupérer le mot de passe en clair de l'utilisateur depuis CAS. Ce mot de passe peut ensuite être fourni à toute portlet demandant l'attribut utilisateur "*password*".

Aussi la mise en place de cette extension du protocole CAS se fait :

- sur CAS
- et sur Esup-uPortal

Pour CAS, nous vous référons à ces documentations : https://wiki.jasig.org/display/casum/clearpass - si vous utilisez memcached, cette page est à consulter également : https://wiki.jasig.org/display/CASUM/ClearPass+and+Multiple+Server+Configurations

Pour uPortal, nous vous référons à cette documentations : https://wiki.jasig.org/display/UPM40/Caching+and+Re-playing+Credentials

Un exemple de mise en place du clearPass sur une portlet peut être esup-filemanager montant des espaces windows en cifs avec une authentification clearPass.

Blocs des configs uPortal pour prise en compte du clearPass

-> Git diff depuis un Esup-uPortal 4.15

```
diff --git a/uportal-war/src/main/resources/properties/contexts/portletContainerContext.xml b/uportal-war/src
/main/resources/properties/contexts/portletContainerContext.xml
index 5b51eeb..d97e3f8 100644
--- a/uportal-war/src/main/resources/properties/contexts/portletContainerContext.xml
+++ b/uportal-war/src/main/resources/properties/contexts/portletContainerContext.xml
@@ -114,10 +114,10 @@
      | NOTE:  Other configuration is also needed to obtain and cache the user's password so this bean can
provide it.
      | See the uPortal manual https://wiki.jasig.org/display/UPM40/Caching+and+Re-playing+Credentials.
      -->
-    <!--bean id="cachedPasswordUserInfoService"
+    <bean id="cachedPasswordUserInfoService"
         class="org.jasig.portal.portlet.container.services.CachedPasswordUserInfoService">
         <property name="decryptPassword" value="false"/>
-    </bean-->
+    </bean>

     <bean id="portalAdministrationService" class="org.apache.pluto.driver.container.
DefaultPortalAdministrationService">
        <property name="administrativeRequestListeners">
diff --git a/uportal-war/src/main/resources/properties/portal.properties b/uportal-war/src/main/resources
/properties/portal.properties
index f1b4028..16ded0c 100644
--- a/uportal-war/src/main/resources/properties/portal.properties
+++ b/uportal-war/src/main/resources/properties/portal.properties
@@ -457,7 +457,7 @@ org.jasig.portal.portlets.googleWebSearch.search.result.type=googleCustom
 ## PORTAL_HOME (see applicationContext.xml).  This is used to encrypt the user's password stored in-memory in
the
 ## security context so malicious code or a hacker is less likely to obtain user's credentials.
 ##
-org.jasig.portal.portlets.passwordEncryptionKey=changeme
+org.jasig.portal.portlets.passwordEncryptionKey=esup_changeme_encryption-key

 ##
 ## Duration in milliseconds between attempts to retrieve the user's password from CAS ClearPass (if enabled).
diff --git a/uportal-war/src/main/resources/properties/security.properties b/uportal-war/src/main/resources
/properties/security.properties
index 19aef54..60b64a7 100644
--- a/uportal-war/src/main/resources/properties/security.properties
+++ b/uportal-war/src/main/resources/properties/security.properties
@@ -46,8 +46,8 @@
 ## This is the factory that supplies the concrete authentication class
 root=org.jasig.portal.security.provider.UnionSecurityContextFactory
-root.cas=org.jasig.portal.security.provider.cas.CasAssertionSecurityContextFactory
+root.cas=org.jasig.portal.security.provider.cas.clearpass.PasswordCachingCasAssertionSecurityContextFactory
 root.simple=org.jasig.portal.security.provider.SimpleSecurityContextFactory

 ## principalToken and crednetialToken declare what tokens
@@ -95,7 +95,7 @@ authorizationProvider=org.jasig.portal.security.provider.AuthorizationServiceFac
 org.jasig.portal.channels.CLogin.CasLoginUrl=https://cas.univ-ville.fr/login

 ## URL of the CAS clearPass password service
-#org.jasig.portal.security.provider.cas.clearpass.PasswordCachingCasAssertionSecurityContextFactory.
clearPassCasUrl==${environment.build.cas.protocol}://${environment.build.cas.server}${environment.build.cas.
context}/clearPass
+org.jasig.portal.security.provider.cas.clearpass.PasswordCachingCasAssertionSecurityContextFactory.
clearPassCasUrl=https://cas.univ-ville.fr/clearPass

 ##
 ##  Local Only Authentication
```

Blocs des configs Jasi-CAS pour mise en place de l'extension clearPass

Ce diff correspond à la mise en place de l'extension clearPass sur un CAS 3.5.2.1 et configuré initialement en utilisant Memcached pour stocker les tickets CAS.
De fait on configure ici clearPass en utilisant Memcached pour stocker les passwords (sous forme hashée/encryptée au niveau du memcached).

```
diff --git a/.project b/.project
index d9361f4..f8771ca 100644
--- a/.project
+++ b/.project
@@ -11,19 +11,13 @@
                        </arguments>
                </buildCommand>
                <buildCommand>
-                       <name>org.maven.ide.eclipse.maven2Builder</name>
-                       <arguments>
-                       </arguments>
-               </buildCommand>
-               <buildCommand>
                        <name>org.eclipse.m2e.core.maven2Builder</name>
                        <arguments>
                        </arguments>
                </buildCommand>
        </buildSpec>
        <natures>
-               <nature>org.eclipse.m2e.core.maven2Nature</nature>
                <nature>org.eclipse.jdt.core.javanature</nature>
-               <nature>org.maven.ide.eclipse.maven2Nature</nature>
+               <nature>org.eclipse.m2e.core.maven2Nature</nature>
        </natures>
 </projectDescription>
diff --git a/cas-server-webapp/pom.xml b/cas-server-webapp/pom.xml
index 22d2903..f3d8329 100644
--- a/cas-server-webapp/pom.xml
+++ b/cas-server-webapp/pom.xml
@@ -203,7 +203,15 @@
        <artifactId>cas-server-integration-memcached</artifactId>
        <version>${project.version}</version>
        <type>jar</type>
-    </dependency>
+    </dependency>
+
+    <dependency>
+        <groupId>org.jasig.cas</groupId>
+        <artifactId>cas-server-extension-clearpass</artifactId>
+        <version>${project.version}</version>
+        <type>jar</type>
+    </dependency>
+

   </dependencies>

diff --git a/cas-server-webapp/src/main/webapp/WEB-INF/deployerConfigContext.xml b/cas-server-webapp/src/main
/webapp/WEB-INF/deployerConfigContext.xml
index a03381b..1d1f5a5 100644
--- a/cas-server-webapp/src/main/webapp/WEB-INF/deployerConfigContext.xml
+++ b/cas-server-webapp/src/main/webapp/WEB-INF/deployerConfigContext.xml
@@ -53,7 +53,7 @@
                class="org.jasig.cas.authentication.AuthenticationManagerImpl">

                <!-- Uncomment the metadata populator to allow clearpass to capture and cache the password
-                    This switch effectively will turn on clearpass.
+                    This switch effectively will turn on clearpass. -->
                <property name="authenticationMetaDataPopulators">
                   <list>
                      <bean class="org.jasig.cas.extension.clearpass.CacheCredentialsMetaDataPopulator">
@@ -61,7 +61,6 @@
                      </bean>
                   </list>
                </property>
-                -->

                <!--
                        | This is the List of CredentialToPrincipalResolvers that identify what Principal is
trying to authenticate.
```

```
@@ -269,4 +268,21 @@
       </list>
     </property>
   </bean>
+
+
+  <bean id="clearPassProxyList" class="org.jasig.cas.client.validation.ProxyList">
+      <constructor-arg>
+          <list>
+              <value>https://ent.univ-ville.fr/uPortal/CasProxyServlet</value>
+              <value>https://ent1.univ-ville.fr/uPortal/CasProxyServlet</value>
+              <value>https://ent2.univ-ville.fr/uPortal/CasProxyServlet</value>
+              <value>https://ent3.univ-ville.fr/uPortal/CasProxyServlet</value>
+              <value>https://ent4.univ-ville.fr/uPortal/CasProxyServlet</value>
+          </list>
+      </constructor-arg>
+  </bean>
+
+
 </beans>
diff --git a/cas-server-webapp/src/main/webapp/WEB-INF/spring-configuration/clearpass-configuration.xml b/cas-
server-webapp/src/main/webapp/WEB-INF/spring-configuration/clearpass-configuration.xml
new file mode 100644
index 0000000..3afb508
--- /dev/null
+++ b/cas-server-webapp/src/main/webapp/WEB-INF/spring-configuration/clearpass-configuration.xml
@@ -0,0 +1,116 @@
+<?xml version="1.0" encoding="UTF-8"?>
+<!--
+
+    Licensed to Jasig under one or more contributor license
+    agreements. See the NOTICE file distributed with this work
+    for additional information regarding copyright ownership.
+    Jasig licenses this file to you under the Apache License,
+    Version 2.0 (the "License"); you may not use this file
+    except in compliance with the License.  You may obtain a
+    copy of the License at the following location:
+
+      http://www.apache.org/licenses/LICENSE-2.0
+
+    Unless required by applicable law or agreed to in writing,
+    software distributed under the License is distributed on an
+    "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
+    KIND, either express or implied.  See the License for the
+    specific language governing permissions and limitations
+    under the License.
+
+-->
+<beans xmlns="http://www.springframework.org/schema/beans"
+       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
+       xmlns:p="http://www.springframework.org/schema/p"
+       xmlns:sec="http://www.springframework.org/schema/security"
+       xmlns:util="http://www.springframework.org/schema/util"
+       xsi:schemaLocation="
+       http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util.xsd
+       http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-
3.1.xsd
+       http://www.springframework.org/schema/security http://www.springframework.org/schema/security/spring-
security-3.1.xsd">
+
+
+    <bean id="CPserialTranscoder" class="net.spy.memcached.transcoders.SerializingTranscoder"
+              p:compressionThreshold="2048" />
+
+    <bean id="credentialsCache" class="net.spy.memcached.CacheMap">
+        <constructor-arg index="0">
+            <bean class="net.spy.memcached.spring.MemcachedClientFactoryBean"
+                  p:servers="localhost:11211"
+                  p:protocol="BINARY"
+                  p:locatorType="ARRAY_MOD"
+                  p:failureMode="Redistribute"
```

```
+                  p:transcoder-ref="CPserialTranscoder">
+                <property name="hashAlg">
+                    <util:constant static-field="net.spy.memcached.DefaultHashAlgorithm.FNV1A_64_HASH" />
+                </property>
+            </bean>
+        </constructor-arg>
+        <constructor-arg index="1" value="7200" /> <!-- this is the timeout for the cache in seconds -->
+        <constructor-arg index="2" value="clearPass_" /> <!-- this is the prefix for the keys stored in the
map -->
+    </bean>
+
+
+    <!--
+        NOTE:
+        Name of delegated ticket registry bean in ticketRegistry.xml must be "ticketRegistryValue."
+    -->
+    <bean id="ticketRegistry" class="org.jasig.cas.extension.clearpass.TicketRegistryDecorator">
+        <constructor-arg index="0" ref="ticketRegistryValue"/>
+        <constructor-arg index="1" ref="credentialsCache"/>
+    </bean>
+
+
+  <!-- implementation of the clear pass vending service -->
+  <bean id="clearPassController" class="org.jasig.cas.extension.clearpass.ClearPassController">
+    <constructor-arg index="0" ref="credentialsCache"/>
+  </bean>
+
+  <bean id="handlerMappingClearPass" class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping"
+        p:alwaysUseFullPath="true">
+    <property name="mappings">
+      <props>
+        <prop key="/clearPass">
+          clearPassController
+        </prop>
+      </props>
+    </property>
+  </bean>
+
+  <!-- Security configuration -->
+  <bean id="clearPassFilterChainProxy" class="org.springframework.security.web.FilterChainProxy">
+    <sec:filter-chain-map request-matcher="ant">
+      <sec:filter-chain pattern="/clearPass"
+                        filters="casValidationFilter,httpServletRequestWrappingFilter"/>
+    </sec:filter-chain-map>
+  </bean>
+  <!-- NOTE:
+         It is dangerous to include a non-proxied CAS Filter for protecting /clearPass. Non-proxied CAS Filters
+         like AuthenticationFilter don't honor the Filter chain proxy protection mechanism and, worse yet,
allow access to the
+         logged on user's cleartext password. It could be useful to enable this bean for easy testing of
clearPass functionality however.-->
+   <!--
+  <bean id="casAuthenticationFilter" class="org.jasig.cas.client.authentication.AuthenticationFilter">
+    <property name="casServerLoginUrl" value="${cas.securityContext.casProcessingFilterEntryPoint.loginUrl}"/>
+    <property name="serverName" value="${server.name}"/>
+  </bean>
+  -->
+  <!--
+    NOTE:
+    A bean named clearPassProxyList must be defined in deployerConfigContext.xml that defines
+    the list of proxying services authorized to obtain clearpass credentials.
+  -->
+  <bean id="casValidationFilter" class="org.jasig.cas.client.validation.
Cas20ProxyReceivingTicketValidationFilter">
+    <property name="serverName" value="${server.name}"/>
+    <property name="exceptionOnValidationFailure" value="false"/>
+    <property name="useSession" value="true"/>
+    <property name="ticketValidator">
+      <bean class="org.jasig.cas.client.validation.Cas20ProxyTicketValidator">
+        <constructor-arg index="0" value="${server.prefix}" />
+        <property name="allowedProxyChains" ref="clearPassProxyList" />
```

```
+      </bean>
+    </property>
+  </bean>
+
+  <bean id="httpServletRequestWrappingFilter" class="org.jasig.cas.client.util.HttpServletRequestWrapperFilter"
/>
+
+</beans>
diff --git a/cas-server-webapp/src/main/webapp/WEB-INF/web.xml b/cas-server-webapp/src/main/webapp/WEB-INF/web.
xml
index 642feb2..ed8ae29 100644
--- a/cas-server-webapp/src/main/webapp/WEB-INF/web.xml
+++ b/cas-server-webapp/src/main/webapp/WEB-INF/web.xml
@@ -55,6 +55,17 @@
     <url-pattern>/services/*</url-pattern>
   </filter-mapping>

+
+  <filter>
+      <filter-name>clearPassFilterChainProxy</filter-name>
+      <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
+  </filter>
+  <filter-mapping>
+      <filter-name>clearPassFilterChainProxy</filter-name>
+      <url-pattern>/clearPass</url-pattern>
+  </filter-mapping>
+
+
   <filter>
     <filter-name>characterEncodingFilter</filter-name>
     <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
@@ -213,7 +224,12 @@
       <servlet-name>cas</servlet-name>
       <url-pattern>/clearauth</url-pattern>
   </servlet-mapping>
-
+
+  <servlet-mapping>
+      <servlet-name>cas</servlet-name>
+      <url-pattern>/clearPass</url-pattern>
+  </servlet-mapping>
+
   <session-config>
     <!-- Default to 5 minute session timeouts -->
     <session-timeout>5</session-timeout>
```