

Second TP : des applets simples.

Vous pouvez consulter les cours suivants :

C1 : <http://www-lipn.univ-paris13.fr/~gayral/CoursGraphique/coursApplet05-06.pdf>

C2 : http://deptinfo.unice.fr/~grin/messupports/java/Securite_fichiers/frame.html

1 – Ecrire une classe applet Affiche (classe qui hérite de Applet) dont la méthode Paint affiche deux chaînes : "Début de l'applet" et "Fin de l'applet". La méthode init ne fait rien.

Ecrire ensuite un fichier index.html qui utilise l'applet (avec avant et après deux messages quelconques). Pour cela (C1 transparents 25 et 26) on utilise la balise APPLET avec son attribut code : `<APPLET CODE="MonApplet.class" WIDTH="800" HEIGHT="300">`

On peut tester avec appletviewer dans un premier temps.

Puis déployer comme un composant web, en plaçant le fichier html, la classe compilée à la racine de l'archive et enfin le répertoire WEB-INF (avec web-inf basique), et y accéder en local à partir d'un navigateur. Essayer ensuite à distance par exemple sous une session Windows.

Changer ensuite une des chaînes dans l'applet puis la recharger : vous pouvez avoir besoin de faire afficher la console Java pour nettoyer le chargeur de classes (ou fermer totalement le navigateur).

2 – Utiliser la console Java pour afficher la version de java et l'OS du client.

(System.getProperty : on peut faire une recherche sur la documentation Java).

3 – Ecrire une première page index.jsp qui teste si un paramètre nommé *nom* lui est transmis. Si le paramètre n'a pas de valeur, afficher un formulaire qui saisit le nom puis rejoindre la même page. Si le paramètre a une valeur, poursuivre sur le chargement d'une applet qui affiche "Bienvenue *nom*". Dans l'applet, la méthode getParameter (de la classe Applet) permet de récupérer la valeur (String) associée à un paramètre.

4 – Introduction sécurité : on cherche à accéder à une information privée du client

A – Utiliser la console Java pour afficher le nom de l'utilisateur client.

B – Après avoir constaté le problème de permission d'accès, on peut signer l'applet :

a- on génère une clé : `keytool -genkey -alias un_alias -keystore .keystore`

b- on construit une archive qui contient l'applet à signer : `jar cf monApplet.jar monApplet.class`

c- on signe l'applet en utilisant la clé :

`jarsigner -keystore .keystore -signedjar monAppletSignee.jar monApplet.jar un_alias`

d- on ajoute à la balise applet du fichier html l'attribut `archive = monAppletSignee.jar`

e- on inclut dans le war à déployer la version signée du jar de l'applet plutôt que la classe Applet.

Lorsqu'on accepte le certificat, l'applet disposera des mêmes droits que n'importe quel exécutable sur le client.