

Sécurité Java

Objectif :

Le langage Java contient intrinsèquement de nombreux mécanismes permettant l'élaboration de programme sûr. Ces mécanismes concernent les différentes facettes de la sécurité comme l'intégrité, la confidentialité, l'identification sûre la protection contre les malveillances.

Cette formation Sécurité Java permet de passer en revue ces différents sujets et propose à chaque fois des ateliers pédagogiques permettant de comprendre en profondeur les mécanismes d'exécution de la JVM. Enfin, le dernier chapitre décrit les utilisations de ses mécanismes dans les applications Java EE

Participants :

Cette formation Sécurité Java s'adresse à des programmeurs, développeurs, chefs de projet désireux de maîtriser JSF à des fins opérationnelles.

Pré-requis :

Il est demandé aux participants de connaître les notions de base du langage Java.

Travaux pratiques :

Les travaux pratiques utilise l'IDE Eclipse.

Durée : 2 jours (14 heures) **Référence : JSJA**

Contenu de la formation :

Introduction et rappels

Ce chapitre introduit le stage en rappelant les enjeux de la sécurité et les réponses intrinsèques du langage Java.

Chargement et vérification des classes

Un des premiers mécanismes implémentés par la JVM est de s'assurer que le code chargé en mémoire ne peut contourner les mécanismes de protection du langage.

Rôle du compilateur Java

Rôle des classloader

Les différentes zones mémoires de la JVM et leur gestion par le garbage collector

Hierarchie des différents classloader

Vérification du byte-code

Chargement dynamique ce classe

Implémenter un class loader

TRAVAUX PRATIQUES :

Modification d'un fichier .class et exécution avec l'option -noverify, Implémentation d'un classloader

chargeant des classes cryptées

Gestionnaire de sécurité et permissions

Les permissions et le gestionnaire de sécurité permettent de contrôler finement les autorisations d'un programme et ses interactions avec son environnement.

Opérations contrôlables

Activation du gestionnaire de sécurité

Domaine de protection, provenance du code et permissions

Parcours de l'API

Fichier .policy

Les classes Permission

Implémentation d'une classe Permission

TRAVAUX PRATIQUES :

Mise au point d'un fichier .policy, implémentation d'une classe Permission

JAAS, Authentification et Autorisations

JAAS offre les services d'authentification et d'autorisations des utilisateurs ou systèmes externes interagissant avec l'application tout en restant du indépendant de la technologie d'authentification.

Présentation de JAAS

LoginContext et LoginModule

Configuration et empilement des login modules

LoginModule disponibles

Implémentation d'un login module spécifique, les CallbackHandler

Packaging d'un login module

Autorisations, Objet Subject et Principals

Interface PrivilegedAction

Configuration des permissions

TRAVAUX PRATIQUES :

Implémentation d'un LoginModule, Configuration des autorisations à partir de rôles utilisateurs.

Signatures numériques et chiffrement

Les techniques de cryptographie permettent de garantir la provenance d'un code et la sa non altération.

Empreinte de message, SHA1 et MD5

Signature numérique, clé publiques et clés privées

L'outil keytool et les keystore

L'outil jarsigner

Les autorités de certification

Déploiement de code signé dans un intranet ou sur internet

Permissions basées sur des keystore

Chiffrement de données, les algorithmes AES et RSA

TRAVAUX PRATIQUES :

Vérification d'une empreinte, Déploiement d'un applet dans un intranet, Chiffrement symétrique et asymétrique

Application de la sécurité dans un environnement Web

Ce chapitre expose l'application des concepts précédents à l'environnement Web exposés.

Sécurisation d'un serveur applicatif Java
Authentification des utilisateurs, descripteur de déploiement d'une application web
Configuration des logins module dans les principaux serveurs applicatifs
Sécurité déclarative des différents tiers de Java EE
SSL.

TRAVAUX PRATIQUES :

Sécurisation d'une application web