

SECURITYMANAGER CHECKPACKAGEDEFINITION METHOD

http://www.tutorialspoint.com/java/lang/securitymanager_checkpackagedefinition.htm Copyright © tutorialspoint.com

Description

The **java.lang.SecurityManager.checkPackageDefinitionStringpkg** method throws a **SecurityException** if the calling thread is not allowed to define classes in the package specified by the argument. This method is used by the **loadClass** method of some class loaders.

This method first gets a list of restricted packages by obtaining a comma-separated list from a call to **java.security.Security.getProperty "package.definition"**, and checks to see if **pkg** starts with or equals any of the restricted packages. If it does, then **checkPermission** gets called with the **RuntimePermission "defineClassInPackage. " + pkg** permission. If this method is overridden, then **super.checkPackageDefinition** should be called as the first line in the overridden method.

Declaration

Following is the declaration for **java.lang.SecurityManager.checkPackageDefinition** method

```
public void checkPackageDefinition(String pkg)
```

Parameters

- **pkg** -- the package name.

Return Value

This method does not return a value.

Exception

- **SecurityException** -- if the calling thread does not have permission to define classes in the specified package.

Example

Our examples require that the permissions for each command is blocked. A new policy file was set that allows only the creating and setting of our Security Manager. The file is in **C:/java.policy** and contains the following text:

```
grant {  
    permission java.lang.RuntimePermission "setSecurityManager";  
    permission java.lang.RuntimePermission "createSecurityManager";  
    permission java.lang.RuntimePermission "usePolicy";  
};
```

The following example shows the usage of **lang.SecurityManager.checkPackageDefinition** method.

```
package com.tutorialspoint;  
  
public class SecurityManagerDemo extends SecurityManager {  
  
    // checkPackageDefinition needs to be overridden  
    @Override  
    public void checkPackageDefinition(String pkg) {  
        throw new SecurityException();  
    }  
  
    public static void main(String[] args) {  
  
        // set the policy file as the system security policy  
        System.setProperty("java.security.policy", "file:C:/java.policy");  
    }  
}
```

```
// create a security manager
SecurityManagerDemo sm = new SecurityManagerDemo();

// set the system security manager
System.setSecurityManager(sm);

// perform the check
sm.checkPackageDefinition("com.tutorialspoint");

// print a message if we passed the check
System.out.println("Allowed!");
}
```

Let us compile and run the above program, this will produce the following result:

```
Exception in thread "main" java.lang.SecurityException
Loading [MathJax]/jax/output/HTML-CSS/jax.js
```