CMER

Centre for Mobile Education and Research

BlackBerry APIs

Week III

Overview

- Blackberry APIs
- Controlled APIs
- Registering to use RIM APIs
- Code Signing
- Optional Signatures
- Code Signature Verification
- Signing Limitations

BlackBerry APIs

- Research In Motion (RIM) tracks the use of sensitive APIs in the BlackBerry Java Development Environment for security and export control reasons.
- These sensitive APIs are known as controlled APIs.
- If you use these classes or methods in your applications, the .cod files must be digitally signed by RIM before you can load them onto handhelds.

Controlled APIs

- Three categories of Research In Motion (RIM) Controlled APIs exist:
  - Runtime APIs
  - BlackBerry Application APIs
  - BlackBerry Cryptography APIs
- BlackBerry Java Applications that use controlled APIs in the BlackBerry Device Simulator can be executed without code signatures however, you must request code signatures from RIM before you are able to load these BlackBerry Java Applications on BlackBerry devices

Controlled APIs (Cont.)

- **The following API packages requires code signatures before you can load it on a BlackBerry device:**
  - **net.rim.blackberry.api.browser**
    - **This package enables applications to invoke the BlackBerry Browser**
  - **net.rim.blackberry.api.invoke**
    - **This package enables applications to invoke BlackBerry applications, such as tasks, messages, MemoPad and phone**

Controlled APIs (Cont.)

- net.rim.blackberry.api.mail
  - This package enables applications to interact with the BlackBerry messages application to send, receive, and open email messages
- net.rim.blackberry.api.mail.event
  - This package defines messaging events and listener interfaces to manage mail events.
- net.rim.blackberry.api.menuitem
  - This package enables you to add custom menu items to BlackBerry applications, such as the address book, calendar, and messages.

Controlled APIs (Cont.)

- net.rim.blackberry.api.options
  - This package enables you to add items to the handheld options
- net.rim.blackberry.api.pdap
  - This package enables applications to interact with BlackBerry personal information management (PIM) applications, including address book, tasks, and calendar. Most of the same functionality is provided by the MIDP package javax.microedition.pim.

Controlled APIs (Cont.)

- net.rim.blackberry.api.phone
  - This package provides access to advanced features of the phone application.
- net.rim.blackberry.api.phone.phonelogs
  - This package provides access to the phone call history.
- net.rim.device.api.browser.field
  - This package enables applications to display a browser field within their user interface

Controlled APIs (Cont.)

- net.rim.device.api.browser.plugin
  - This package enables you to add support for additional MIME types to the BlackBerry Browser.
- net.rim.device.api.crypto.*
  - These packages provide data security capabilities, including data encryption and decryption, digital signatures, data authentication, and certificate management
- net.rim.device.api.io.http
  - This package enables applications to register with the BlackBerry Browser as provider for one or more URLs.

Controlled APIs (Cont.)

- net.rim.device.api.notification
  - This package provides methods to trigger event notifications and respond to system-wide and application-specific events
- net.rim.device.api.servicebook
  - This package enables applications to add, delete, and access service book entries.
- net.rim.device.api.synchronization
  - This package enables applications to perform backup and restore operations on custom data.

Controlled APIs (Cont.)

- net.rim.device.api.system
  - This package provides classes that enable functionality such as persistent data storage, interprocess communication (IPC), SMS, network communication using datagrams, and application management.

Register to use RIM controlled APIs

1. Complete the registration form on the BlackBerry® Developer Zone at https://www.blackberry.com/JDEKeys.
2. Save the .csi file that Research In Motion (RIM) sends to you in an email message. The .csi file contains a list of signatures and your registration information. If the BlackBerry Signing Authority Tool administrator does not provide you with the .csi file or the Client PIN and you are an ISV partner, contact your ISV Technical Partnership Manager. If you are not an ISV partner, send an email message to jde@rim.com.

Register to use RIM controlled APIs (Cont.)

3. Double-click the .csi file.
4. If a dialog box appears that states that a private key cannot be found, follow the instructions to create a new key pair file.
5. In the Registration PIN field, type the PIN that RIM provided.

Register to use RIM controlled APIs (Cont.)

6. In the Private Key Password field, type a password of at least eight characters. The private key password protects your private key. If you lose this password, you must register again with RIM. If this password is stolen, contact RIM immediately.
7. Click Register.
8. Click Exit.

Code signing request process

1. **The Signature Tool opens an HTTP connection to the signing authority system and sends a request.**
   - **The Signature Tool sends an SHA-1 hash of your codein the .csl and .cso files so that the signing authority system can generate the necessary signature**
   - **Your actual code is not sent to RIM.**

Code signing request process (Cont.)

2. The signing authority system verifies that the request is valid and applies a RIM private key to the hash of each .cod file to create the signatures.
3. The signing authority system returns the signatures to the Signature Tool and closes the HTTP connection.
4. The Signature Tool appends the signatures to each .cod file.

Code signing request process (Cont.)

- When the files are signed, the Status column for the .cod file displays Signed.
- If any problems occur with the signature request, the Status column displays Failed

- When your .cod files are signed, you can load them onto the BlackBerry Wireless Handheld.

Optional Signatures

- You can load applications onto handhelds without optional .cso signatures.
- These signatures are only required if their corresponding methods are invoked during runtime.
- When the application calls a method that requires a signature, the VM verifies that the application has this authorization.
- If the VM does not find these optional signatures, the application stops

Request code signatures

- **After you obtain a .csi file from Research In Motion (RIM), you are able to request code signatures.**
  1. **In Windows® Explorer, locate the .cod file for the BlackBerry® Java® Application for which you are requesting a signature.**
  2. **Make sure that a .csl file with the same name as the .cod file exists in the same folder as the .cod file. The BlackBerry Integrated Development Environment compiler automatically generates the .csl file.**

Request code signatures (Cont.)

3. Double-click the .cod file to add it to the signature list. The signature list contains information on the .cod files that you want permission to access and are requesting signatures for.
4. Repeat steps 1 through 3 for each .cod file that you want to add to the signature list.
5. On the BlackBerry Signature Tool menu, click Request. The BlackBerry Signature Tool is part of the BlackBerry Java® Development Environment installation. The BlackBerry JDE is available for download from the BlackBerry Developer Zone: http://www.blackberry.com/developers/

Request code signatures (Cont.)

6. In the dialog box, type your private key password.
7. Click OK. The BlackBerry Signature Tool uses the private key password to append the signature to the request, and it sends the signature list of .cod files to the Web Signer application for verification. The Web Signer application installs when you install the BlackBerry Signing Authority Tool. See the BlackBerry Signing Authority Tool Version 1.0 - Password Based Administrator Guide for more information about the Web Signer application.

Request code signatures using a proxy server (Cont.)

- When requesting code signatures using a proxy server, two tasks are required:
  - Register signature keys using a proxy server.
  - Sign a BlackBerry Java Application using a proxy server.
- Register signature keys using a proxy server.
  - You can register each .csi file only once.
    1. At the command prompt, browse to the BlackBerry® Signature Tool bin directory. For example:

Register signature keys using a proxy server

- C:\Program Files\Research In Motion\BlackBerry JDE 4.3.1\bin

2. Type the following command:

- Java -jar -Dhttp.proxyHost=myproxy.com -Dhttp.proxyPort=80 SignatureTool.jar SigKey.csi
  - » SigKey: The name of each signature key (.csi) file. Use the following naming conventions for the keys: client-RRT-*.csi, client-RBB-*.csi, client-RCR-*.csi.
  - » Dhttp.proxyHost: The name or IP address of the proxy server.
  - » Dhttp.proxyPort: The proxy server port number if you do not specify 80 as the default port number.

Register signature keys using a proxy server (Cont.)

3. **Repeat step 2 for each .csi file that you want to register.**

Sign a BlackBerry Java Application using a proxy server

1. At the command prompt, browse to the BlackBerry Signature Tool bin directory. For example:
   - C:\Program Files\Research In Motion\BlackBerry JDE 4.3.1\bin
2. Type the following command:
   - Java -jar -Dhttp.proxyHost=myproxy.com -Dhttp.proxyPort=80 SignatureTool.jar
3. In the File Selection window, select the .cod file(s) to sign.
4. Click Open.

View signature status

1. Start the BlackBerry® Signature Tool.
2. Select a .cod file.
3. View the Status column:
   - For files the Web Signer has signed, the Status column contains Signed.
   - For files the Web Signer did not sign, the Status column contains Failed. The Web Signer might have rejected the .cod file because the private key password was typed incorrectly

Code signature verification

- There are two types of code signature verification:
  - Linktime verification
    - When you load a signed .cod file onto the handheld, the virtual machine (VM) links the .cod file with the API libraries and verifies that the .cod file includes the required signatures. If a signature is missing, the VM stops linking and does not load the application.

Code signature verification (Cont.)

- **Runtime verification**
  - **When the user uses the application on the handheld, if the application invokes a method that requires a signature, the VM verifies that the application contains the necessary signature. If the signature is not present, a ControlledAccessException is thrown and the requested operation is not performed.**

Signing limitations

- There are several situations in which the code signing process does not proceed.
- The signing authority administrator can limit your access to signatures by specifying a limit using both time and frequency parameters. These parameters are defined in your .csi file. Be aware of these possible limitations when applying for signatures.

Lost data

- You cannot perform any code signing requests without your .csi file. Your registration key is stored within your .csi file.
- None of your signature requests can be sent to the signing authority system if the Signature Tool cannot find this key and sign your requests with it.

Lost data (Cont.)

- If your system stops responding, and you lose data or even entire file structures, you might discover that you have also lost the ability to perform signing requests.
- If you lose your .csi file, the Signature Tool cannot communicate with the signing authority system on your behalf.
- If you lose your .csi file, contact your signing authority administrator and request a new one