REALIZE ANY EXPECTED SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF BUSINESS OPPORTUNITY, OR CORRUPTION OR LOSS OF DATA, FAILURES TO TRANSMIT OR RECEIVE ANY DATA, PROBLEMS ASSOCIATED WITH ANY APPLICATIONS USED IN CONJUNCTION WITH RIM PRODUCTS OR SERVICES, DOWNTIME COSTS, LOSS OF THE USE OF RIM PRODUCTS OR SERVICES OR ANY PORTION THEREOF OR OF ANY AIRTIME SERVICES, COST OF SUBSTITUTE GOODS, COSTS OF COVER, FACILITIES OR SERVICES, COST OF CAPITAL, OR OTHER SIMILAR PECUNIARY LOSSES, WHETHER OR NOT SUCH DAMAGES WERE FORESEEN OR UNFORESEEN, AND EVEN IF RIM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREBIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR OF ANY REMEDY CONTAINED HEREBIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR SUCCESSORS, ASSIGNS, AGENTS, SUPPLIERS (INCLUDING AIRTIME SERVICE PROVIDERS), AUTHORIZED RIM DISTRIBUTORS (ALSO INCLUDING AIRTIME SERVICE PROVIDERS) AND THEIR RESPECTIVE DIRECTORS, EMPLOYEES AND INDEPENDENT CONTRACTORS.

IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

Prior to subscribing for, installing or using any Third Party Products and Services it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers may not offer Internet browsing functionality with a subscription to BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM’s products and services may require one or more patent, trademark, copyright or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use, Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM’s products and services are provided as a convenience to you and are provided "AS IS" with no express or implied conditions, endorsements, guarantees, representations or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server software, BlackBerry® Desktop Software, and/or BlackBerry® Device Software and may require additional development or Third Party Products and Services for access to corporate applications.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/) and/or licensed pursuant to Apache License, Version 2.0 (http://www.apache.org/licenses/). For more information, see the NOTICE.txt file included with the software. 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.
The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.
Chapter 9: Developing secure applications

This chapter outlines security concerns related to Java® applications for BlackBerry® smartphones. It defines cryptography, and describes the steps you can take to encrypt communication and authenticate users, including the steps you need to take to manage certificates.

In this lab, you will use the Cryptography API to encrypt and decrypt files, use symmetric keys, public/private key pairs, and AES encryption. You will test your application using the BlackBerry® Smartphone Simulator.
Chapter 9: Developing secure applications

9.1 Encryption and decryption

In this exercise, you will build an API that will provide encryption and decryption using the Cryptography API and a symmetric key pair.

1. Double-click the BasicCryptoDeveloperLab_1.zip.

2. In the IDE, open the BasicCryptoDeveloperLab.jdw workspace.

3. Open the BasicCryptoDeveloperLab.java source file.

4. Complete the implementation of the encrypt method:
   a. Create a new AES symmetric key using keyData.
   b. Create a new EncryptorEngine and FormatterEngine using the newly created key.
   c. Create a BlockEncryptor using the FormatterEngine.
   d. Create a new Digest (SHA-1) and hash the data using the digest.
   e. Encrypt the data and append the hash returning the ciphertext.

5. Complete the implementation of the decrypt method:
   a. Create a new AES symmetric key using keyData.
   b. Create a new DecryptorEngine and UnformatterEngine using the newly created key.
   c. Create a BlockDecryptor using the UnformatterEngine.
   d. Decrypt the data.
   e. Create a new Digest (SHA-1) and hash the data using the digest.
   f. Verify that the hash is correct and return the plaintext.

6. Check your work by testing it using the BlackBerry Smartphone Simulator.
9.2 Advanced features of the Cryptography API

In this exercise, you will build an application that uses advanced features in the Cryptography API such as encryption, decryption, signing and verification using public key cryptography. In this exercise you will

- Discover how to use a Public Key for encryption.
- Discover how to use a Private Key for signing.
- Discover how to use a Private Key for decryption.
- Discover how to use a Public Key for verification.

Prerequisite: Complete the Encryption and Decryption exercise.

1. Open AdvancedCryptoDeveloperLab_1.zip
2. In the IDE, open the AdvancedCryptoDeveloperLab.jdw workspace
3. Open the AdvancedCryptoDeveloperLab.java source file
4. Complete the implementation of the encrypt method:
   1. Implement the encrypt method using the RSAPublicKey.
   2. Implement the sign method using the RSAPrivateKey.
   3. Implement the decrypt method using the RSAPrivateKey.
   4. Implement the verify method using the RSAPublicKey.
      Note: You must import the net_rim_crypto.jar file into your project to access the Cryptography API.
5. Check your work by comparing it to the contents of AdvancedCryptoDeveloperLab_2.zip
9.3 Implementing AES encryption and decryption

In this exercise, you will edit a Java application source file to implement AES encryption and decryption on the BlackBerry smartphone.

1. Open the CryptoDemo.java source file.
2. Locate the line of code where key is defined as a TripleDES key.
3. Modify the code so the key is an AES key.
4. Locate the line of code where encryptionEngine is defined as a TripleDES encryption engine.
5. Modify the code so the encryption engine is an AES encryption engine.
6. Locate the line of code where decryptorEngine is defined as a TripleDES decryptor engine.
7. Modify the code so the decryptor engine is an AES decryptor engine.
9.4 Enhancing encryption and decryption

In this exercise, you will edit a Java application source file to further increase data security on the BlackBerry smartphone. You will implement the AESKey class using a random byte array.

1. Open the CryptoDemo.java source file.

2. Above the AESKey definition, add a line of code that creates a byte array called keyData.

3. Assign keyData an initial value using the RandomSource class. Use the getBytes method to return 256 random bytes.

4. Modify the AESKey definition so it uses the keyData array.

5. Check your work by building and testing the application.