DRD17-J. Do not use the Android cryptographic security provider encryption default for AES

(THIS CODING RULE OR GUIDELINE IS UNDER CONSTRUCTION)

The predominant Android cryptographic security provider API defaults to using an insecure AES encryption method: ECB block cipher mode for AES encryption. Android's default cryptographic security provider (since version 2.1) is BouncyCastle.

NOTE: Java also chose ECB as a default value when only the AES encryption method is chosen. So, this rule also applies to Java, but for Java's different default cryptographic security provider. Oracle Java's default cryptographic security provider is SunJCE.

Noncompliant Code Example

This noncompliant code example shows an application that ..., and hence not secure.

Compliant Solution

In this compliant solution ...

Related Guidelines

| The CERT Oracle Secure Coding Standard for Java | DRD18-J. Do not use the default behavior in a cryptographic library if it does not use recommended practices |

Risk Assessment

If an insecure encryption method is used, then the encryption does not assure privacy, integrity, and authentication of the data.

<table>
<thead>
<tr>
<th>Rule</th>
<th>Severity</th>
<th>Likelihood</th>
<th>Remediation Cost</th>
<th>Priority</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRD17-J</td>
<td>High</td>
<td>Likely</td>
<td>Medium</td>
<td>P18</td>
<td>L1</td>
</tr>
</tbody>
</table>

Automated Detection

Automatic detection of ...

Bibliography

| Egele 2013 | An Empirical Study of Cryptographic Misuse in Android Applications |