NUM03-J. Use integer types that can fully represent the possible range of unsigned data

The only unsigned primitive integer type in Java is the 16-bit char data type; all of the other primitive integer types are signed. To interoperate with native languages, such as C or C++, that use unsigned types extensively, any unsigned values must be read and stored into a Java integer type that can fully represent the possible range of the unsigned data. For example, the Java long type can be used to represent all possible unsigned 32-bit integer values obtained from native code.

Noncompliant Code Example

This noncompliant code example uses a generic method for reading integer data without considering the signedness of the source. It assumes that the data read is always signed and treats the most significant bit as the sign bit. When the data read is unsigned, the actual sign and magnitude of the values may be misinterpreted.

```java
public static int getInteger(DataInputStream is) throws IOException {
    return is.readInt();
}
```

Compliant Solution

This compliant solution requires that the values read are 32-bit unsigned integers. It reads an unsigned integer value using the readInt() method. The readInt() method assumes signed values and returns a signed int; the return value is converted to a long with sign extension. The code uses an & operation to mask off the upper 32 bits of the long, producing a value in the range of a 32-bit unsigned integer, as intended. The mask size should be chosen to match the size of the unsigned integer values being read.

```java
public static long getInteger(DataInputStream is) throws IOException {
    return is.readInt() & 0xFFFFFFFFL; // Mask with 32 one-bits
}
```

As a general principle, you should always be aware of the signedness of the data you are reading.

Risk Assessment

Treating unsigned data as though it were signed produces incorrect values and can lead to lost or misinterpreted 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>NUM03-J</td>
<td>Low</td>
<td>Unlikely</td>
<td>Medium</td>
<td>P2</td>
<td>L3</td>
</tr>
</tbody>
</table>

Automated Detection

Automated detection is infeasible in the general case.

Bibliography

| [API 2006] | Class DataInputStream: method readInt |
| [Harold 1997] | Chapter 2, "Primitive Data Types, Cross-Platform Issues, Unsigned Integers" |
| [Hitchens 2002] | Section 2.4.5, "Accessing Unsigned Data" |
| [Seacord 2015] | NUM03-J. Use integer types that can fully represent the possible range of unsigned data LiveLesson |