**Code Correctness: Non-Static Inner Class Implements Serializable**

**Development Mitigation SOP**

Code correctness vulnerabilities occur when an Object API is not used properly or as intended.

Code correctness vulnerabilities can occur when a non-static inner class implements Serializable because it may leak information from the outer class or lead to a runtime error if the outer class is not Serializable. This issue may also cause platform dependencies since the Java compiler creates synthetic fields in order to implement inner classes, but these are implementation dependent, and may vary from compiler to compiler.

**Defense Against Code Correctness: Non-Static Inner Class Implements Serializable**

Inner classes should not be serialized or they should be changed to static-nested classes because static-nested classes do not have the drawbacks that non-static inner classes have when serialized. A static nested class has no association with instance variables and do not cause serialization of the outer class.

**Examples**

**General Example**

…

class User implements Serializable {

private int accessLevel;

class Registrator implements Serializable {

…

}

}

**Explanation**

The code above shows the inner class, Registrator, as Serialized, but it will also serialize the field accessLevel from the outer class User*.*

**Recommendation**

The code below stops the inner class from implementing java.io.Serializable:

…

class User implements Serializable {

private int accessLevel;

class Registrator {

…

}

}

**Example**

private class EditAction extends AbstractAction {…}

**Explanation**

The code above shows the inner class, AbstractAction, as Serialized.

**Recommendation**

The code below stops the inner class from implementing java.io.Serializable:

private class EditAction{

class AbstractAction {…}

}

**Resources**

1. [HP Enterprise Security – Code Correctness: Non-Static Inner Class Implements Serializable](https://vulncat.fortify.com/en/detail?id=desc.structural.java.code_correctness_non_static_inner_class_implements_serializable#Java%2fJSP)