**Obsolete Development Mitigation SOP**

Obsolete vulnerabilities occur when the use of deprecated or obsolete functions indicate neglected code. As programming languages evolve, methods can become obsolete due to advances in the language, improvements in understanding of how operations should perform effectively/securely, and changes in conventions. Just because a function may be deprecated or replaced does not necessarily mean they pose a security risk. However, it is often an indication that the code surrounding the function may be neglected and need to be updated.

**Defense Against Obsolete**

Deprecated or obsolete functions should be replaced with modern counterparts. It is also important to examine the code around the obsolete function to analyze whether there are any larger maintenance problems with the code in that area.

**Examples**

**General Example**

String name = new String(nameBytes, highByte);

**Explanation**

This code constructs a string object from an array of bytes and a value that specifies the top 8 bits of each 16-bit Unicode character. This constructor may fail to correctly convert bytes to characters depending upon which charset is used to encode the string, nameBytes*,* due to the evolution of the charsets used to encode strings. This constructor was deprecated and replaced by a constructor that accepts the name of the *charset* used to encode the bytes for conversion.

**Example**

public void createDBQTaskMetric(long claimId, DevelopmentPlanTask developmentPlanTask, SystemUser currentUser) {

try {

systemEventService.**notifyObservers**(new

TaskAddedToDevPlanEvent( claimId, developmentPlanTask,

currentUser ));

} catch(Exception e) {

LOGGER.error(“Couldn’t get claim for metric.”, e);

}

}

**Explanation**

This finding was flagged on the notifyObservers() method (bolded above).

**Recommendation**

The next step would be to find the notifyObservers() method definition in the SystemEventService.java class which is shown below:

@Deprecated

void notifyObservers(SystemEvent event);

Since the method is deprecated, this indicated neglected code and the method needs to be updated.

**References**

1. [HP Enterprise Security – Obsolete](http://www.hpenterprisesecurity.com/vulncat/en/vulncat/java/obsolete.html)