

# Fortify SCA Rulepacks Version 2015.4.0.0008 Java Scanning Issues (Related To Increased False Positives)

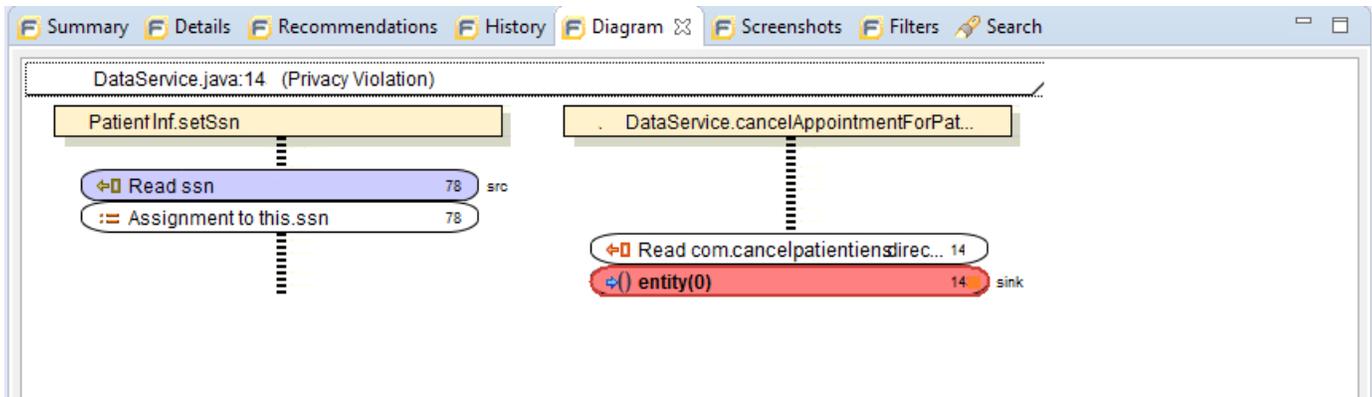
The purpose of this announcement is to share with the VA developer and contractor community an observation of an increase in the number of false positive findings reported in Java applications using latest release of the HP Fortify SCA rulepacks, and also to share VA Software Assurance Program Office instructions how to handle it. Specifically with regards to the latter, in short, they still must be audited per VA Secure Code Review SOP.

The observed false positives are in HP Fortify SCA version 4.40 using the 2015.4.0.0008 rulepacks, when scanning Java applications. *Other programming languages are not affected by this issue.* Several applications have reported that after scanning their source code using the latest rulepacks, they've seen many more critical and high findings reported.

Types of findings that are showing an increase in the number of false positives include:

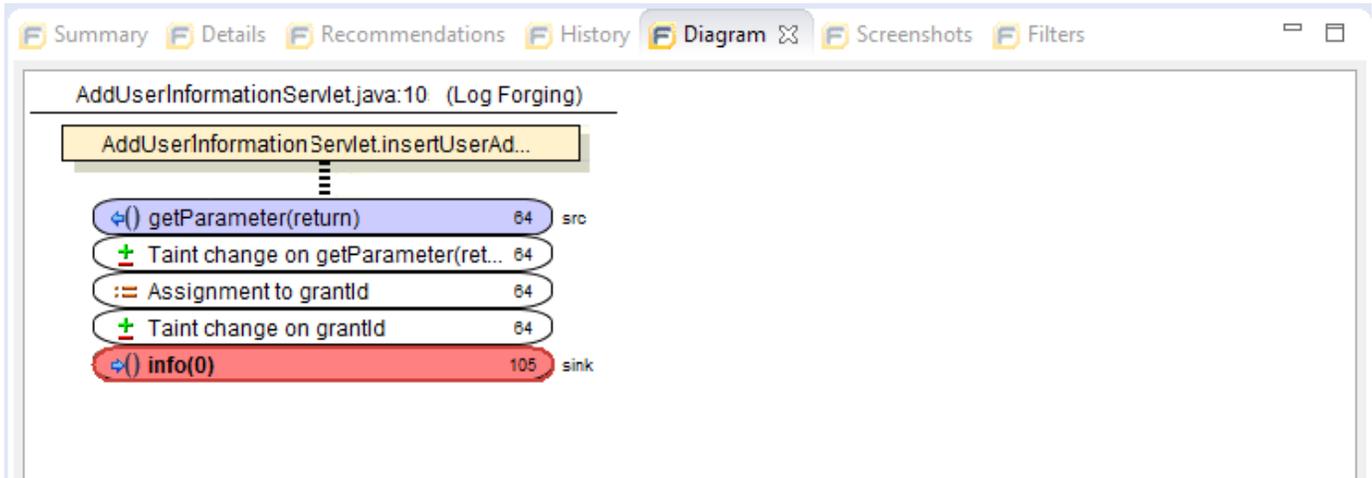
- Privacy Violation
- SQL Injection
- SQL Injection: Persistence
- Log Forging

Other types may also be affected, but this is generally only being observed in data flow types of findings. These false positives may be identified by viewing the Diagram tab in Audit Workbench for each finding. Generally, Fortify will show a flow of control for the source of the finding (highlighted in blue) to the left of the window. To the right, the flow of control for the sink of the finding (highlighted in red) is shown. When there is no correlation between the source and the sink, and you can verify in the code that the source Fortify has flagged is not used in the output identified at the sink, this is a false positive. To illustrate:



In the above Diagram, an ssn is read in one portion of the code, as shown to the left. In the right, the code is building data for output, but this output does not include the ssn, so this is a False Positive.

To illustrate a valid finding:



In this Diagram, data is being read from the request object, identified by the source. At the sink, the same data is being output, with no validation having occurred before the data is output, so this is a true positive finding. Note that many diagrams will be far more complex, as data is transferred to different locations in the code. If you can observe that the data, identified in the source, is actually used in the data output at the sink, with no validation of that data occurring, this is a true positive finding and must be fixed, or comments added explaining other mitigating factors in the system.

VA developers should audit findings that follow the false positive scenario, as shown in the first diagram, as "Not an Issue," with comments added stating, "The source of this finding, is not used in the output, or sink, shown by Fortify.", in accordance with VA Secure Code Review SOP (found [here](#)).