

Conventional scanners yield too many false positives

Conventional code scanners identify suspicious code by a technique called *pattern matching*. They can find all instances of a defined ABAP statement, e.g. DELETE DATASET.

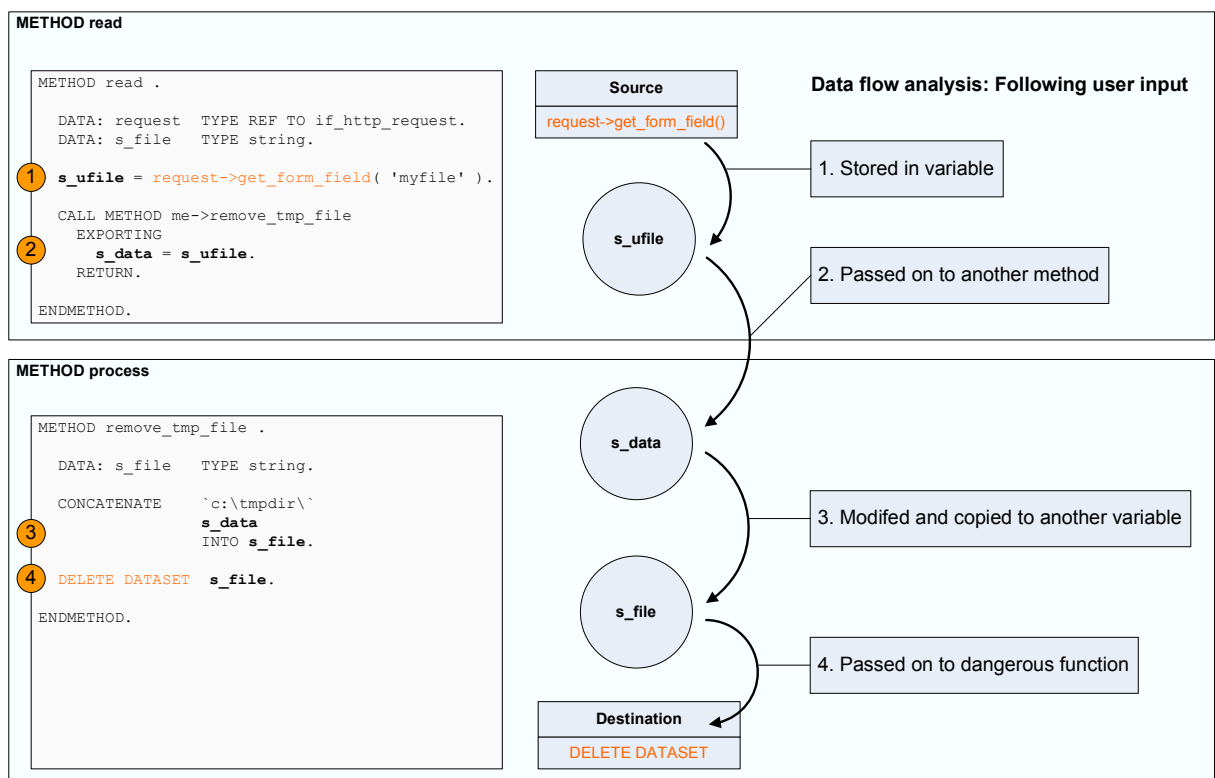
But the key concern in security is to determine, if a potentially dangerous statement is *exploitable*: Can attackers actually reach and misuse the dangerous statement?

To stay with our example: Deleting a file on the server is not dangerous itself. But it becomes dangerous if an unauthorized user can actually control which file is to be deleted.

Trying to identify all instances where user input reaches a DELETE DATASET statement with pattern matching is a lot of work. It means reviewing every single result and manually analyzing corresponding ABAP code. This approach is far from static code analysis.

The solution is data flow analysis

Data flow analysis is a technique that identifies sources and potential destinations of input. It then analyzes, if there are any connections between the sources and destinations. Any identified connection (data flow) indicates that the finding is most likely exploitable.



CODEPROFILER uses data flow analysis in combination with a comprehensive rule set with dozens of data sources and dangerous destinations.

Summary

- Only data flow analysis yields meaningful and reliable test results for security.
- Without data flow analysis, significant human interaction is necessary to validate the tool's findings.