Changes in results post upgrade - 8.3.31

Summary: this page lists:

- Impacts of changes made to CAST AIP 8.3.31 on Quality Model results post upgrade
- Other impacts of changes made in CAST AIP 8.3.31

All changes in results related to extensions are now listed in the extension documentation and will not appear in this page.

Impacts of changes made in CAST AIP 8.3.31 on Quality Model results post upgrade

Mainframe

Check PCB status code after DLI queries - 8160

The rule "Check PCB status code after DLI queries" (https://technologies.castsoftware.com/rules?s=8160|qualityrules|8160) has been modified to improve functionality. As a result of these changes your results may be impacted after upgrade.

Variables defined in Working-Storage section must be initialized before to be read - 8034

The rule "Variables defined in Working-Storage section must be initialized before to be read" (https://technologies.castsoftware.com/rules?s=8034|qualityrules|8034) has been modified to improve functionality. As a result of these changes your results may be impacted after upgrade.

Other impacts of changes made in CAST AIP 8.3.31

Mainframe

Missing links from JCL to Cobol Program when CALL syntax is used

Missing links from JCL to Cobol Program when CALL syntax is used. This is now fixed and after an upgrade your existing results may be impacted.

User Input Security

Improved support for .NET uncontrolled string format

User Input Security is now more precisely able to detect Uncontrolled string format vulnerabilities for .NET source code. As a consequence, some false positive violations reported when using previous releases of AIP Core may be removed after upgrade.

Improved support for the detection of SQL injections in applications using the Entity Framework for .NET

The methods SqlQuery and ExecuteSqlCommandAsync are now considered as database targets for SQL injection. System.Data.Find methods are no longer considered as database targets for SQL injection. As a result of these changes your results may be impacted after upgrade.

Improved support for the detection of SQL injections in applications using the Oracle framework for .NET

The methods ExecuteNonQuery(), ExecuteReader(), ExecuteReader([System.Data]System.Data.CommandBehavior), ExecuteScalar() and ExecuteStream() are now considered as database targets for SQL injection. As a result of these changes your results may be impacted after upgrade.

Improved support for the detection of SQL injections in applications using the Dapper framework for .NET

Methods such as QueryAsync, QueryFirstAsync, QueryFirstOrDefaultAsync etc. from the Dapper framework are now considered as database targets for SQL injection. As a result of these changes your results may be impacted after upgrade.

Improved support for the detection of SQL injections in applications using the logging framework System.Diagnostics.Trace for .NET

Methods such as TraceInformation, TraceWarning, TraceError etc. from the logging framework System.Diagnostics.Trace for .NET are now considered as database targets for SQL injection. As a result of these changes your results may be impacted after upgrade.

New support for the detection of deserialization injections in applications using the YAML framework for .NET

The methods Load([]System.IO.TextReader) and Load([]YamlDotNet.Core.IParser) from the YAML framework for .NET are now considered as targets for deserialization injection. As a result of these changes your results may be impacted after upgrade.

New support for the detection of deserialization injections in applications using the XStream framework for JEE

fromXML type methods from the XStream framework for JEE are now considered as targets for deserialization injection. As a result of these changes your results may be impacted after upgrade.