OVAL Reporting

OVAL Reporting (formerly OCRL) seeks to address the case where security relevant information can be automatically located and extracted from a system, but where this information requires a human to determine whether it complies with policies. Towards this end, OVAL Reporting automatically extracts system state and then formats it into a report that can be reviewed by a user at a later time.

In June of 2009, MITRE presented an initial draft of the Open Checklist Reporting Language (OCRL) at Security Automation Developer Days. The general consensus at this gathering was that OCRL's capabilities represented a valuable addition to security benchmarks, but that it should make use of structures in the OVAL language when possible rather than creating new structures.

Based on this input, MITRE designed OVAL Reporting. The new design places the OVAL Report Template schema in a separate namespace within the OVAL language (similar to the way that the System Characteristics and Results schemas have their own namespace). As such, it integrates with and utilizes existing OVAL structures (e.g. OVAL Objects) but causes no changes to any of these structures. No content or tools that process OVAL Definitions, System Characteristics, or Results are affected by the addition of the new namespace to OVAL.

OVAL Reporting uses OVAL structures to guide the data gathering portion of its functionality. In addition, to further simplify use, the reporting section of the language has also been re-engineered and now uses XSLT to organize and format reports. This use of well-known, existing technologies is intended to minimize the new structures authors need to learn in order to utilize the language. To assist with common actions undertaken in the creation of a report, OVAL Reporting also includes a library of functions usable within XSLT documents.

Processing Model

OVAL Reporting envisions a two-part processing model. In the first part, an OVAL engine extracts the OVAL structures from an OVAL Report Template. These OVAL structures are processed to produce a System Characteristics (SC) file. (The file might either be on disk or in memory.) The OVAL engine then enters the second phase in which the reporting instructions are extracted and a standard XSLT Processor applies them to the SC file produced in part one to produce a report. Figure 1 summarizes this process.

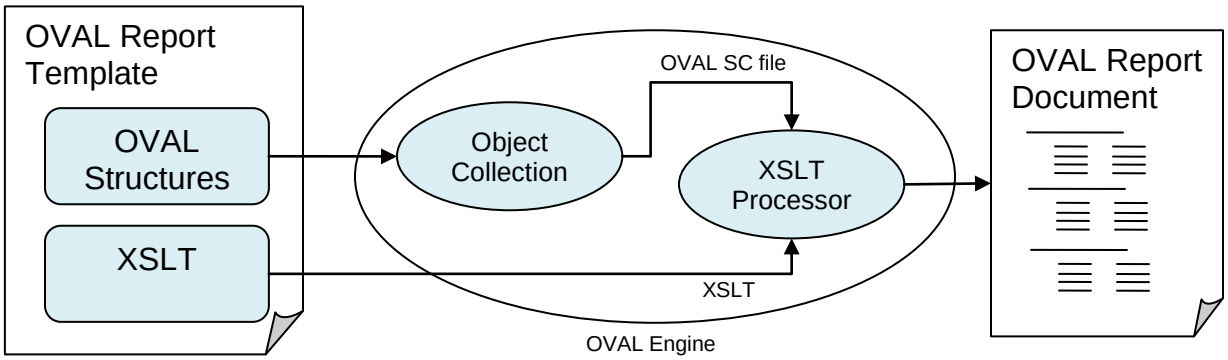

Figure 1. Basic Processing Model

An alternate processing model is also envisioned where, instead of generating a fresh SC file, an existing SC file could be referenced to format reports based on prior OVAL assessments.

Request for Comments

OVAL Reporting duplicates the capabilities of the previous OCRL language, but does so using pre-existing language components. OVAL Report Template authors only need to be familiar with the construction of OVAL and XSLT in order to create structures that will collect system information and compile it into a report.

We are in the process of drafting out the details of the schema itself. As we do that, however, it would be useful to hear feedback from the community regarding the general shape of the new design. Comments, concerns, and suggestions regarding the new language are welcome.