SWAMP Summary File

James A. Kupsch

Version 1.0.0, 2017-01-31 10:41:52 CST

Table of Contents

1. Introduction

This document describes the SWAMP summary file. The SWAMP summary file is a dataset containing summary information from assessments run in the SWAMP including both tools that report weaknesses found in the source code and those that report metrics about the source code. Each assessment summary is associated with a package, an operating system platform, and an assessment tool.

There are two types of summary data depending on the type of tool. For tools that produce weakness results, the summary includes the count of the weakness types reported along with CWE identifiers if available. For tools that produce code metric results such as lines of code or complexity, the summary includes the count of number of objects (files or functions/methods) the metric was reported, and the sum, minimum, maximum, sum of the squares of the values, average and standard deviation.

The summary file can contain summaries from any number of packages and assessments. Only assessments that were successful are included in the summary data. If the package failed to build or be assessed, it is not included in the data set.

Listing 1 is an example SWAMP summary file that contains the essential elements and structure. A typical summary file would be larger as it would include multiple packages assessed on multiple platfoms with multiple tools. It would also typically have more than one weakness type or code metric.

The SWAMP summary file uses an XML format. The rest of this document presents the structure and meaning of the XML elements and attributes of a SWAMP summary file.

Listing 1. An example SWAMP summary file containing one package and results from the assessments with four analysis tools, two weakness type tools (one found no weaknesses) and two code metric type tools (one containing per file metrics and the other containing per function/method level metrics).

```xml
<?xml version="1.0" encoding="UTF-8"?>
<swamp-summaries version="1.0" ts="2017-01-25T16:50:40Z">
 <swamp-summary>
  <package name="PACKAGE_NAME" version="1.1">
   <platform name="ubuntu-16.04-64">
    <tool name="TOOL_NAME1" version="2.3.4">
     <bug-summaries>
      <bug-summary group="GROUP_NAME" code="CODE_NAME" cwe='398" count="6"/>
     </bug-summaries>
    </tool>
    <tool name="TOOL_NAME2" version="4.2.1"/>
    <tool name="cloc" version="1.70">
     <metric-summaries type="per-file">
      <metric-summary>
       <type>comment-lines</type>
       <count>1</count>
       <sum>0</sum>
       <sum-of-squares>0</sum-of-squares>
       <minimum>0</minimum>
       <maximum>0</maximum>
       <average>0.00</average>
       <standard-deviation>0.00</standard-deviation>
      </metric-summary>
     </metric-summaries>
    </tool>
    <tool name="lizard" version="1.12.7">
     <metric-summaries type="per-function">
      <metric-summary>
       <type>code-lines</type>
       <count>3</count>
       <sum>27</sum>
       <sum-of-squares>629</sum-of-squares>
       <minimum>file</minimum>
       <maximum>25</maximum>
       <average>9.00</average>
       <standard-deviation>13.89</standard-deviation>
      </metric-summary>
     <metric-summaries>
    </tool>
   </platform>
  </package>
 </swamp-summary>
</swamp-summaries>
```

2. <swamp-summaries>

The `<swamp-summaries>` element is the top level element and is required. It contains two attributes:

version

    The version of the SWAMP summary docunent format. Currently 1.0.

ts

    Timestamp of the documents creation in ISO-8601 format. An example value of *2017-01-02T13:02:00Z* denotes January 2, 2017 at 13:02:00 in the UTC timezone. The literal '*T*' separates the date from the time, and the literal '*Z*' denotes the UTC timezone.

3. <swamp-summary>

The `<swamp-summary>` element contains one or more `<package>` elements.

4. <package>

The `<package>` element represents a SWAMP package. All enclosing elements refer to this package. The package has a name and a version specified by two attributes:

name

    The name of the package.

version

    The version of the package.

A `<package>` element contains one or more `<platform>` elements.

5. <platform>

The `<platform>` element represents a SWAMP operating system platform All enclosing elements refer to this platform. The platform has a name that encodes the operating system name, its version, and if it is a 32-bit or 64-bit version (represented by *32* and *64* respectively) separated by a hyphens. The value *ubuntu-16.04-64* represents the *Ubuntu 16.04 64-bit* operating system. The name is specified by the following attribute:

name

    The name of the package. Current SWAMP operating system names are *rhel* (Red Hat Enterprise Linux), *scientific* (Scientific Linux), *centos, fedora, debian*, and *ubuntu*.

A `<platform>` element contains one or more `<tool>` elements.

6. <tool>

The `<tool>` element represents a SWAMP tool. All enclosing elements refer to this tool. A tool has a name and a version. The tool name and version are specified by the following attributes:

name

    The name of the tool.

version

    The version of the tool.

Depending on the type of the tool, a `<tool>` element will contain one or more `<bug-summaries>` or `<metric-summaries>` elements. Current tools will contain one or the other.

7. <bug-summaries>

The `<bug-summaries>` element contains a set of `<bug-summary>` elements for the assessment of the package on the platform using the tool.

8. <bug-summary>

The `<bug-summary>` element is the summary of the number of occurrences of a particular type of weakness produced by the tool during an assessment. The weakness is specified by the combination of the *group* and *code* attributes. The values are specified by attributes of the element:

group

    The tool specific string representing the *group* of the weakness and may be the empty string.

code

    The tool specific string representing the *code* of the weakness and may be the empty string.

count

    The number of occurrences of the *group* and *code* produced by assessing the package using the tool.

cwe

    The CWE (Common Weakness Enumeration) Identifier for the weakness. The CWE Identifier is produced by the tool or from an external mapping. Some tools do not specify CWE values or only produce them for a subset of the weaknesses. The external mapping is not complete, so the CWE value may not be known. If the CWE identifier is not known the *cwe* attribute will not be present.

9. <metric-summaries>

The `<metric-summaries>` element contains a set of `<metric-summary>` elements for the assessment of the package on the platform using the tool. Some metrics are counts per file while others are counts

per function. These are distinguished by the *type* attribute:

type

This required attribute distinguishes the scope of the metric to be a file or a function and is one of the following values:

per-file

The enclosed metric summaries are per file.

per-function

The enclosed metric summaries are per function.

10. <metric-summary>

The `<metric-summary>` element is the summary of a particular code metric produced during an assessment by a metric tool such as lines of code (LOC) or cyclomatic complexity. A code metric is a numeric value, so the summary data is more complex than a `<bug-summary>`. The summary data is contained in child elements and includes the type of the metric, the count (number of times produced), and the sum, minimum, maximum, sum of the squares, average and standard deviation of the values. These are described next:

10.1. <type>

The *type* element determines the meaning of the metric's value. The current values and their definitions are as follows:

code-lines

Number of lines in the file or function containing code as defined by the tool.

blank-lines

Number of lines in the file or function that are empty or only contain whitespace.

comment-lines

Number of lines in the file or function containing comments (excluding lines that contain both code and comments) as defined by the tool.

total-lines

Total number of the lines in the file or function.

ccn

Cyclomatic complexity of the function.

token

Number of tokens in the file or function.

params

Number of parameters to a function.

10.2. <count>

The value of the element is the number of the metric type produced by the tool during an assessment. For *per file* metrics this value is the number of files analyzed, and for *per function* metrics this value is the number of function analyzed.

10.3. <sum>

The value of the element is the sum of the values for the metric type produced by the tool during an assessment.

10.4. <sum-of-squares>

The value of the element is the sum of the square of the values for the metric type produced by the tool during an assessment.

10.5. <minimum>

The value of the element is the minimum value for the metric type produced by the tool during an assessment.

10.6. <maximum>

The value of the element is the maximum value for the metric type produced by the tool during an assessment.

10.7. <average>

The value of the element is the average of the values for the metric type produced by the tool during an assessment.

10.8. <standard-deviation>

The value of the element is the standard deviation of the values for the metric type produced by the tool during an assessment.