- Open Vulnerability and Assessment Language - Element Dictionary

- Schema: Solaris Definition
- Version: 5.2
- Release Date: 31 January 2007

The following is a description of the elements, types, and attributes that compose the Solaris specific tests found in Open Vulnerability and Assessment Language (OVAL). Each test is an extension of the standard test element defined in the Core Definition Schema. Through extension, each test inherits a set of elements and attributes that are shared amongst all OVAL tests. Each test is described in detail and should provide the information necessary to understand what each element and attribute represents. This document is intended for developers and assumes some familiarity with XML. A high level description of the interaction between the different tests and their relationship to the Core Definition Schema is not outlined here.

The OVAL Schema is maintained by The Mitre Corporation and developed by the public OVAL Community. For more information, including how to get involved in the project and how to submit change requests, please visit the OVAL website at http://oval.mitre.org.

< isainfo_test >

The isainfo test reveals information about the instruction set architectures. This information can be retrieved by the isainfo command. It extends the standard TestType as defined in the oval-definitions-schema and one should refer to the TestType description for more information. The required object element references an isainfo_object and the optional state element specifies the metadata to check. The evaluation of the test is guided by the check attribute that is inherited from the TestType.

The isainfo_test was originally developed by Robert L. Hollis at ThreatGuard, Inc. Many thanks for their support of the OVAL project.

Child Elements	Type	MinOccurs	MaxOccurs
object	oval-def:ObjectRefType	1	1
state	oval-def:StateRefType	0	1

< isainfo_object >

The isainfo_object element is used by an isainfo test to define those objects to evaluated based on a specified state. There is actually only one object relating to isainfo and this is the system as a whole. Therefore, there are no child entities defined. Any OVAL Test written to check isainfo will reference the same isainfo_object which is basically an empty object element.

< isainfo_state >

The isainfo_state element defines the information about the instruction set architectures. Please refer to the individual elements in the schema for more details about what each represents.

Child Elements	Type	MinOccurs	MaxOccurs
bits	oval-def:EntityStateStringType	0	1
kernel_isa	oval-def:EntityStateStringType	0	1
application_isa	oval-def:EntityStateStringType	0	1

< package_test >

The package test is used to check information associated with different packages installed on the system. The information used by this test is modeled after the /usr/bin/pkginfo command. It extends the standard TestType as defined in the oval-definitions-schema and one should refer to the TestType description for more information. The required object element references an inetd_object and the optional state element specifies the information to check. The evaluation of the test is guided by the check attribute that is inherited from the TestType.

Child Elements	Type	MinOccurs	MaxOccurs
object	oval-def:ObjectRefType	1	1
state	oval-def:StateRefType	0	1

< package_object >

The package_object element is used by a package test to define the packages to be evaluated. Each object extends the standard ObjectType as definied in the oval-definitions-schema and one should refer to the ObjectType description for more information. The common set element allows complex objects to be created using filters and set logic. Again, please refer to the description of the set element in the oval-definitions-schema.

A package object consists of a single pkginst entity that identifies the package to be used.

Child Elements	Type	MinOccurs	MaxOccurs
pkginst	oval-def:EntityObjectStringType	1	1

< package_state >

The package_state element defines the different information associated with packages installed on the system. Please refer to the individual elements in the schema for more details about what each represents.

Child Elements	Type	MinOccurs	MaxOccurs
pkginst	oval-def:EntityStateStringType	0	1
name	oval-def:EntityStateStringType	0	1
category	oval-def:EntityStateStringType	0	1
version	oval-def:EntityStateStringType	0	1
vendor	oval-def:EntityStateStringType	0	1
description	oval-def:EntityStateStringType	0	1

< patch_test >

The patch test is used to check information associated with different patches installed on the system. The information being tested is based off the /usr/bin/showrev -p command. It extends the standard TestType as defined in the oval-definitions-schema and one should refer to the TestType description for more information. The required object element references an inetd_object and the optional state element specifies the information to check. The evaluation of the test is guided by the check attribute that is inherited from the TestType.

Child Elements	Type	MinOccurs	MaxOccurs
object	oval-def:ObjectRefType	1	1
state	oval-def:StateRefType	0	1

< patch_object >

The patch_object element is used by a patch test to define the specific patch to be evaluated. Each object extends the standard ObjectType as definied in the oval-definitions-schema and one should refer to the ObjectType description for more information. The common set element allows complex objects to be created using filters and set logic. Again, please refer to the description of the set element in the oval-definitions-schema.

A patch object consists of a single base entity that identifies the patch to be used.

Child Elements	Type	MinOccurs	MaxOccurs
base	oval-def:EntityObjectIntType	1	1

< patch_state >

The patch_state element defines the different information associated with a specific patch installed on the system. Please refer to the individual elements in the schema for more details about what each represents.

Child Elements	Type	MinOccurs	MaxOccurs
base	oval-def:EntityStateIntType	0	1
version	oval-def:EntityStateIntType	0	1