

- Open Vulnerability and Assessment Language - Element Dictionary

- Schema: Independent Definition
- Version: 5.3
- Release Date: 6/22/2007 11:17:35 AM

The following is a description of the elements, types, and attributes that compose the tests found in Open Vulnerability and Assessment Language (OVAL) that are independent of a specific piece of software. 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>.

< family_test >

The family_test element is used to check the family a certain system belongs to. This test basically allows the high level system types (window, unix, ios, etc.) to be tested. 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 a family_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.

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

< family_object >

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

< family_state >

The family_state element contains a single entity that is used to check the family associated with the system. The family is a high-level classification of system types.

--	--	--	--

Child Elements	Type	MinOccurs	MaxOccurs
family	ind-def:EntityStateFamilyType	0	1

< filemd5_test >

The file md5 test is used to check the md5 associated with a specified file. 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 a filemd5_object and the optional state element specifies the md5 to check. The evaluation of the test is guided by the check attribute that is inherited from the TestType.

This test has been deprecated. You should use the filehash_test instead. This test will be dropped in the major release of OVAL.

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

< filemd5_object >

The filemd5_object element is used by a file test to define the specific file(s) to be evaluated. Each object extends the standard ObjectType as defined 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 file object defines the path and filename of the file(s). In addition, a number of behaviors may be provided that help guide the collection of objects. Please refer to the Filemd5Behaviors complex type for more information about specific behaviors.

Child Elements	Type	MinOccurs	MaxOccurs
behaviors	ind-def:Filemd5Behaviors	0	1
path	oval-def:EntityObjectStringType	1	1
filename	oval-def:EntityObjectStringType	1	1

< filemd5_state >

The filemd5_state element contains entities that are used to check the file path, name, and the md5 associated with a specific file.

--	--	--	--

Child Elements	Type	MinOccurs	MaxOccurs
path	oval-def:EntityStateStringType	0	1
filename	oval-def:EntityStateStringType	0	1
md5	oval-def:EntityStateStringType	0	1

== Filemd5Behaviors ==

The Filemd5Behaviors complex type defines a number of behaviors that allow a more detailed definition of the filemd5_object being specified.

Attributes:

- max_depth n/a (optional -- default='1')
- recurse_direction n/a (optional -- default='none')

< filehash_test >

The file hash test is used to check the hashes associated with a specified file. 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 a filehash_object and the optional state element specifies the different hashes 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

< filehash_object >

The filehash_object element is used by a file hash test to define the specific file(s) to be evaluated. Each object extends the standard ObjectType as defined 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 file object defines the path and filename of the file(s). In addition, a number of behaviors may be provided that help guide the collection of objects. Please refer to the FilehashBehaviors complex type for more information about specific behaviors.

Child Elements	Type	MinOccurs	MaxOccurs

behaviors	ind-def:FilehashBehaviors	0	1
path	oval-def:EntityObjectStringType	1	1
filename	oval-def:EntityObjectStringType	1	1

< filehash_state >

The filehash_state element contains entities that are used to check the file path, name, and the different hashes associated with a specific file.

Child Elements	Type	MinOccurs	MaxOccurs
path	oval-def:EntityStateStringType	0	1
filename	oval-def:EntityStateStringType	0	1
md5	oval-def:EntityStateStringType	0	1
sha1	oval-def:EntityStateStringType	0	1

== FilehashBehaviors ==

The FilehashBehaviors complex type defines a number of behaviors that allow a more detailed definition of the filehash_object being specified.

Attributes:

-
- max_depth n/a (optional -- default='1')
 - recurse_direction n/a (optional -- default='none')

< environmentvariable_test >

The environmentvariable_test element is used to check an environment variable found on the system. 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 a environmentvariable_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.

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

< environmentvariable_object >

The environmentvariable_object element is used by an environment variable test to define the specific environment variable(s) to be evaluated. Each object extends the standard ObjectType as defined 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.

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

< environmentvariable_state >

The environmentvariable_state element contains two entities that are used to check the name of the specified environment variable and the value associated with it.

Child Elements	Type	MinOccurs	MaxOccurs
name	oval-def:EntityStateStringType	0	1
value	oval-def:EntityStateAnyType	0	1

< sql_test >

The sql test is used to check information stored in a database. It is often the case that applications store configuration settings in a database as opposed to a file. This test has been designed to enable those settings to be tested. 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 a wmi_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.

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

< sql_object >

The sql_object element is used by a sql test to define the specific database and query to be evaluated. Connection information is supplied allowing the tool to connect to the desired database and a query is supplied to call out the desired setting. Each object extends the standard ObjectType as defined 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.

--	--	--	--

Child Elements	Type	MinOccurs	MaxOccurs
engine	ind-def:EntityObjectEngineType		
version	oval-def:EntityObjectStringType		
connection_string	oval-def:EntityObjectStringType		
sql	oval-def:EntityObjectStringType		

< sql_state >

The sql_state element contains two entities that are used to check the name of the specified environment variable and the value associated with it.

Child Elements	Type	MinOccurs	MaxOccurs
engine	ind-def:EntityObjectEngineType	0	
version	oval-def:EntityStateStringType	0	
connection_string	oval-def:EntityStateStringType	0	
sql	oval-def:EntityStateStringType	0	
result	oval-def:EntityStateAnyType	0	

< textfilecontent_test >

The textfilecontent_test element is used to check the contents of a text file (aka a configuration file) by looking at individual lines. 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 a textfilecontent_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.

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

< textfilecontent_object >

The textfilecontent_object element is used by a text file content test to define the specific line(s) of a file(s) to be evaluated. Each object extends the standard ObjectType as defined 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.

Child Elements	Type	MinOccurs	MaxOccurs
behaviors	ind-def:TextfilecontentBehaviors	0	1
path	oval-def:EntityObjectStringType	1	1
filename	oval-def:EntityObjectStringType	1	1
line	oval-def:EntityObjectStringType	1	1

< textfilecontent_state >

The textfilecontent_state element contains entities that are used to check the file path and name, as well as the line in question and the value of the specific subexpression.

Child Elements	Type	MinOccurs	MaxOccurs
path	oval-def:EntityStateStringType	0	1
filename	oval-def:EntityStateStringType	0	1
line	oval-def:EntityStateStringType	0	1
subexpression	oval-def:EntityStateAnyType	0	1

== TextfilecontentBehaviors ==

The TextfilecontentBehaviors complex type defines a number of behaviors that allow a more detailed definition of the textfilecontentBehaviors_object being specified.

Attributes:

-
- max_depth n/a (optional -- default='1')
 - recurse_direction n/a (optional -- default='none')

< unknown_test >

An unknown test acts as a placeholder for tests whose implementation is unknown. Any information that is known about the test should be held in the notes child element that is available through the extension of the abstract test element. It extends the standard TestType as defined in the oval-definitions-schema and one should refer to the TestType description for more information. Note that for an unknown test, the required check attribute that is part of the extended TestType should be ignored during evaluation and hence can be set to any valid value.

< variable_test >

The variable test allows the value of a variable to be compared to a defined value. As an example one might use this test to validate that a variable being passed in from an external source falls within a specified range. 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 a variable_object and the optional state element specifies the value 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

< variable_object >

Child Elements	Type	MinOccurs	MaxOccurs
var_ref	ind-def:EntityObjectVariableRefType	1	1

< variable_state >

The variable_state element contains two entities that are used to check the var_ref of the specified variable and the value associated with it.

Child Elements	Type	MinOccurs	MaxOccurs
var_ref	ind-def:EntityStateVariableRefType	0	1
value	oval-def:EntityStateStringType	0	1

< xmlfilecontent_test >

The xmlfilecontent_test element is used to explore the contents of an xml file. This test basically allows specific pieces of an xml document specified using xpath to be tested. 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 a family_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.

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

< xmlfilecontent_object >

The xmlfilecontent_object element is used by a xml file content test to define the specific piece of an xml file(s) to be evaluated. Each object extends the standard ObjectType as defined 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.

Child Elements	Type	MinOccurs	MaxOccurs
behaviors	ind-def:XmlfilecontentBehaviors	0	1
path	oval-def:EntityObjectStringType	1	1
filename	oval-def:EntityObjectStringType	1	1
xpath	oval-def:EntityObjectStringType	1	1

< xmlfilecontent_state >

The xmlfilecontent_state element contains entities that are used to check the file path and name, as well as the xpath used and the value of the this xpath.

Child Elements	Type	MinOccurs	MaxOccurs
path	oval-def:EntityStateStringType	0	1
filename	oval-def:EntityStateStringType	0	1
xpath	oval-def:EntityStateStringType	0	1
value_of	oval-def:EntityStateStringType	0	1

== XmlfilecontentBehaviors ==

The XmlfilecontentBehaviors complex type defines a number of behaviors that allow a more detailed definition of the xmlfilecontentBehaviors_object being specified.

Attributes:

-
- max_depth n/a (optional -- default='1')
 - recurse_direction n/a (optional -- default='none')

== EntityObjectEngineType ==

The EntityObjectEngineType complex type defines a string entity value that is restricted to a set of enumerations. Each valid enumeration is a valid database engine. The empty string is also allowed to support empty element associated with variable references.

--

Value	Description
access	
db2	
cache	
firebird	
firstsql	
foxpro	
informix	
ingres	
interbase	
lightbase	
maxdb	
monetdb	
mimer	
oracle	
paradox	
pervasive	
postgre	
postgre	
sqlbase	
sqlite	
sqlserver	
sybase	

== EntityStateEngineType ==

The EntityStateEngineType complex type defines a string entity value that is restricted to a set of enumerations. Each valid enumeration is a valid database engine. The empty string is also allowed to support empty element associated with variable references.

Value	Description
access	
db2	
cache	

firebird
firstsql
foxpro
informix
ingres
interbase
lightbase
maxdb
monetdb
mimer
oracle
paradox
pervasive
postgre
postgre
sqlbase
sqlite
sqlserver
sybase

== EntityStateFamilyType ==

The EntityStateFamilyType complex type defines a string entity value that is restricted to a set of enumerations. Each valid enumeration is a high-level family of system operating system. The empty string is also allowed to support empty element associated with variable references.

Value	Description
ios	
macos	
unix	
windows	

== EntityObjectVariableRefType ==

The EntityObjectVariableRefType complex type defines a string object entity that has a valid OVAL variable id as the value. The empty string is also allowed to support empty element associated with variable references.

== EntityStateVariableRefType ==

The EntityStateVariableRefType complex type defines a string state entity that has a valid OVAL variable id as the value. The empty string is also allowed to support empty element associated with variable references.