- Open Vulnerability and Assessment Language - Element Dictionary

- Schema: Windows Definition
- Version: 4.2
- Release Date: 2 December 2005

The following is a description of the elements, types, and attributes that compose the Windows 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.

Elements

This section describes all the elements that are found within the schema, starting with the root element. Note that in the tables outlining possible attributes and child elements, square brackets [] means that the item is optional. All complex and simple types, along with attribute groups are described later in this document.

Account Privileges Test

<accountprivileges_test>

The account privileges test looks at the individual privileges and rights associated with the specified account. Each privilege in the data section of the test can accept a boolean value signifying whether the privilege is granted or not.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<accountprivileges_testid="wnt-0"check="all"comment="account has desired privileges">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <account_name>administrator</account_name>
        </object>
        <dataoperation="AND">
                <account_domain>mitre.org</account_domain>
                <sebackupprivilege>true</sebackupprivilege>
                <selockmemoryprivilege>false</selockmemoryprivilege>
                <seshutdownprivilege>true</seshutdownprivilege>
                <seservicelogonright>true</seservicelogonright>
        </data>
</accountprivileges_test>
```

object section

<account_name>

The name of the account to check the privileges and rights of.

Parent Test:	Account Privileges Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<account_domain>

The domain the specified account belongs to.

Parent Test:	Account Privileges Test

Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<account_sid>

The SID of the specified account.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<seassignprimarytokenprivilege>

If this privilege is enabled, it allows a parent process to replace the access token that is associated with a child process.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seauditprivilege>

If this privilege is enabled, it allows a process to generate audit records in the security log. The security log can be used to trace unauthorized system access.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sebackupprivilege>

If this privilege is enabled, it allows the user to circumvent file and directory permissions to back up the system. The privilege is selected only when an application attempts access by using the NTFS backup application programming interface (API). Otherwise, normal file and directory permissions apply.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sechangenotifyprivilege>

If this privilege is enabled, it allows the user to pass through folders to which the user otherwise has no access while navigating an object path in the NTFS file system or in the registry. This privilege does not allow the user to list the contents of a folder; it allows the user only to traverse its directories.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<secreateglobalprivilege>

If this privilege is enabled, it allows the user to create named file mapping objects in the global namespace during Terminal Services sessions.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<secreatepagefileprivilege>

If this privilege is enabled, it allows the user to create and change the size of a pagefile.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string

Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<secreatepermanentprivilege>

If this privilege is enabled, it allows a process to create a directory object in the object manager. It is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode have this privilege inherently.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<secreatetokenprivilege>

If this privilege is enabled, it allows a process to create an access token by calling NtCreateToken() or other token-creating APIs.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sedebugprivilege>

If this privilege is enabled, it allows the user to attach a debugger to any process. It provides access to sensitive and critical operating system components.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seenabledelegationprivilege>

If this privilege is enabled, it allows the user to change the Trusted for Delegation setting on a user or

computer object in Active Directory. The user or computer that is granted this privilege must also have write access to the account control flags on the object.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seimpersonateprivilege>

If this privilege is enabled, it allows the user to impersonate a client after authentication. It is not supported on Windows XP, Windows 2000 SP3 and earlier, or Windows NT.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seincreasebasepriorityprivilege>

If this privilege is enabled, it allows a user to increase the base priority class of a process.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seincreasequotaprivilege>

If this privilege is enabled, it allows a process that has access to a second process to increase the processor quota assigned to the second process.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean

Valid Operators:	equals, not equal

<seloaddriverprivilege>

If this privilege is enabled, it allows a user to install and remove drivers for Plug and Play devices.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<selockmemoryprivilege>

If this privilege is enabled, it allows a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<semachineaccountprivilege>

If this privilege is enabled, it allows the user to add a computer to a specific domain.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<semanagevolumeprivilege>

If this privilege is enabled, it allows a non-administrative or remote user to manage volumes or disks.

Parent Test:	Account Privileges Test
Cardinality:	0-1

Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seprofilesingleprocessprivilege>

If this privilege is enabled, it allows a user to sample the performance of an application process.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seremoteshutdownprivilege>

If this privilege is enabled, it allows a user to shut down a computer from a remote location on the network.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<serestoreprivilege>

If this privilege is enabled, it allows a user to circumvent file and directory permissions when restoring backed-up files and directories and to set any valid security principal as the owner of an object.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sesecurityprivilege>

If this privilege is enabled, it allows a user to specify object access auditing options for individual resources such as files, Active Directory objects, and registry keys. A user who has this privilege can also

view and clear the security log from Event Viewer.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seshutdownprivilege>

If this privilege is enabled, it allows a user to shut down the local computer.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sesyncagentprivilege>

If this privilege is enabled, it allows a process to read all objects and properties in the directory, regardless of the protection on the objects and properties. It is required in order to use Lightweight Directory Access Protocol (LDAP) directory synchronization (Dirsync) services.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sesystemenvironmentprivilege>

If this privilege is enabled, it allows modification of system environment variables either by a process through an API or by a user through System Properties.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean

Valid Operators:	equals, not equal

<sesystemprofileprivilege>

If this privilege is enabled, it allows a user to sample the performance of system processes.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sesystemtimeprivilege>

If this privilege is enabled, it allows the user to adjust the time on the computer's internal clock. It is not required to change the time zone or other display characteristics of the system time.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<setakeownershipprivilege>

If this privilege is enabled, it allows a user to take ownership of any securable object in the system, including Active Directory objects, NTFS files and folders, printers, registry keys, services, processes, and threads.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<setcbprivilege>

If this privilege is enabled, it allows a process to assume the identity of any user and thus gain access to the resources that the user is authorized to access.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seundockprivilege>

If this privilege is enabled, it allows the user of a portable computer to undock the computer by clicking Eject PC on the Start menu.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seunsolicitedinputprivilege>

If this privilege is enabled, it allows the user to read unsolicited data from a terminal device.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sebatchlogonright>

If an account is assigned this right, it can log on using the batch logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seinteractivelogonright>

If an account is assigned this right, it can log on using the interactive logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<senetworklogonright>

If an account is assigned this right, it can log on using the network logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seremoteinteractivelogonright>

If an account is assigned this right, it can log on to the computer by using a Remote Desktop connection.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<seservicelogonright>

If an account is assigned this right, it can log on using the service logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sedenybatchLogonright>

If an account is assigned this right, it is explicitly denied the ability to log on using the batch logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sedenyinteractivelogonright>

If an account is assigned this right, it is explicitly denied the ability to log on using the interactive logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sedenynetworklogonright>

If an account is assigned this right, it is explicitly denied the ability to log on using the network logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sedenyremoteInteractivelogonright>

If an account is assigned this right, it is explicitly denied the ability to log on through Terminal Services.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string

Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<sedenyservicelogonright>

If an account is assigned this right, it is explicitly denied the ability to log on using the service logon type.

Parent Test:	Account Privileges Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

Active Directory Test

<activedirectory_test>

This test gathers information about specified entries in active directory.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<activedirectory_testid="wat-0"check="all"comment="allow execute permissions to HTTP
virtual dir">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <naming_context>configuration</naming_context>
                <relative_dnoperator="pattern match">^CN=[^,]+,CN=[^,]
                +,CN=HTTP,CN=Protocols,CN=[^,]*,CN=Servers,CN=[^,]
                *,CN=Administrative Groups,CN=[^,]*,CN=Microsoft
                Exchange,CN=Services$</relative_dn>
                <attribute>msExchAccessFlags</attribute>
        </object>
```

```
        <dataoperation="AND">
                <adstype>ADSTYPE_INTEGER</adstype>
                <valueoperator="bitwise and">512</value>
        </data>
</activedirectory_test>
```

object section

<naming_context>

Each object in active directory exists under a certain naming context (also known as a partition). A naming context is defined as a single object in the Directory Information Tree (DIT) along with every object in the tree subordinate to it. There are three default naming contexts in Active Directory: domain, configuration, and schema.

Parent Test:	Active Directory Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<relative_dn>

The relative_dn field is used to uniquely identify an object inside the specified naming context. It contains all the parts of the objects distinguished name except those outlined by the naming context.

Parent Test:	Active Directory Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<attribute>

Specifies a named value contained by the object.

Parent Test:	Active Directory Test

Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<object_class>

The name of the class of which the object is an instance.

Parent Test:	Active Directory Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<adstype>

Specifies the type of information that the specified attribute represents.

Parent Test:	Active Directory Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<value>

The actual value of the specified active directory attribute.

Parent Test:	Active Directory Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	binary, boolean, float, int, string
	equals, not equal, greater than, less than, greater than or

Valid Operators:	equal, less than or equal, bitwise and, bitwise or, pattern match

Audit Event Policy Test

<auditeventpolicy_test>

The auditeventpolicy test enumerates the different types of events the system should audit. The defined values are found in window's POLICY_AUDIT_EVENT_TYPE enumeration and accessed through the LsaQueryInformationPolicy when the InformationClass parameters are set to PolicyAuditEventsInformation.

Extends:	standardTestType
Valid Sections:	notes, data

```
<auditeventpolicy_testid="wbt-0"check="all"comment="test certain event policies">
    <oval:notes>
        <oval:note>This is an example test written under version 4 of the OVAL schema. It
        meant to give a short overview of the test and might not contain every possible child
        element.</oval:note>
    </oval:notes>
    <dataoperation="AND">
        <account_logon>AUDIT_FAILURE</account_logon>
        <directory_service_access>AUDIT_SUCCESS_FAILURE</directory_service_acce
    </data>
</auditeventpolicy_test>
```

data section

<account_logon>

Audit attempts to log on to or log off of the system. Also, audit attempts to make a network connection.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1

Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<account_management>

Audit attempts to create, delete, or change user or group accounts. Also, audit password changes.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<detailed_tracking>

Audit specific events, such as program activation, some forms of handle duplication, indirect access to an object, and process exit.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<directory_service_access>

Audit attempts to access the directory service.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<logon>

Audit attempts to log on to or log off of the system. Also, audit attempts to make a network connection.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<object_access>

Audit attempts to access securable objects, such as files.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<policy_change>

Audit attempts to change Policy object rules.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<privilege_use>

Audit attempts to use privileges.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<system>

Audit attempts to shut down or restart the computer. Also, audit events that affect system security or the security log.

Parent Test:	Audit Event Policy Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

File Test

<file_test>

This test checks file metadata. The time information can be retrieved by the _stat function.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<file_testid="wft-0"check="at least one"comment="the version of mshtml.dll is less than 5.1.26(
    <oval:notes>
        <oval:note>This is an example test written under version 4 of the OVAL schema. It
        short overview of the test and might not contain every possible child element.</oval
    </oval:notes>
    <object>
        <pathdatatype="component">
            <componenttype="registry_value">HKEY_LOCAL_MACHINE\SOFTWAR
            NT\CurrentVersion\SystemRoot</component>
            <componenttype="literal">\system32\mshtml.dll</component>
        </path>
    </object>
    <data>
        <versiondatatype="version"operator="less than">
            <major>5</major>
            <minor>1</minor>
            <build>2600</build>
            <private>128</private>
        </version>
    </data>
</file_test>
```

object section

<path>

Specifies the absolute path to a file on the machine. This path can be created from multiple components that are added together. When a pattern match operator is used, the corresponding regular expression is matched against the set of absolute path strings. These string would not include the '.' and '..' notations. This means that a '.*' component of a regular expression will not only match all files in the specified directories, but all subdirectories, their subdirectories, etc.

Parent Test:	File Test
Cardinality:	1
Content:	none
Valid Datatypes:	component
Valid Operators:	equals, not equal, pattern match

data section

<owner>

A string that contains the name of the owner.

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<size>

Size of the file in bytes.

Parent Test:	File Test
Cardinality:	0-1
Content:	integer

Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<a_time>

Time of last access of file. Valid on NTFS but not on FAT formatted disk drives. The string should represent the FILETIME structure which is a 64-bit value representing the number of 100-nanosecond intervals since January 1, 1601 (UTC).

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal, pattern match

<c_time>

Time of creation of file. Valid on NTFS but not on FAT formatted disk drives. The string should represent the FILETIME structure which is a 64-bit value representing the number of 100-nanosecond intervals since January 1, 1601 (UTC).

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal, pattern match

<m_time>

Time of last modification of file. The string should represent the FILETIME structure which is a 64-bit value representing the number of 100-nanosecond intervals since January 1, 1601 (UTC).

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, greater than, less than, greater than or

	equal, less than or equal, pattern match

<ms_checksum>

the md5 checksum of the file.

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<md5>

The md5 hash of the file

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<version>

The version of the file.

Parent Test:	File Test
Cardinality:	0-1
Content:	none
Valid Datatypes:	version
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<type>

The type child element marks wether the file test describes a directory, named pipe, standard file, etc. These types are the return values for GetFileType, with the exception of FILE_ATTRIBUTE_DIRECTORY which is obtained by looking at GetFileAttributesEx.

Parent Test:	File Test

Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<development_class>

The development_class element allows the distinction to be made between the GDR development environment and the QFE development environment. This field holds the text found in front of the mmmmmm-nnnn version, for example srv03_gdr.

Parent Test:	File Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

File Audited Permissions Test

<fileauditedpermissions_test>

This test looks at the audited access rights of a given file that a system access control list (SACL) structure grants to a specified trustee. The trustee's audited access rights are determined checking all access control entries (ACEs) in the SACL. For help with this test see the GetAuditedPermissionsFromAcl() api.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<fileauditedpermissions_testid="wht-0"check="at least one"comment="a file exists with the spec
    <oval:notes>
        <oval:note>This is an example test written under version 4 of the OVAL schema. It
        short overview of the test and might not contain every possible child element.</oval
    </oval:notes>
    <object>
        <path>
```

```
                    <componenttype="registry_value">HKEY_LOCAL_MACHINE\SOFTWAR
                    NT\CurrentVersion\SystemRoot</component>
                    <componenttype="literal">\system32\mshtml.dll</component>
                </path>
                <trustee_name>SYSTEM</trustee_name>
            </object>
            <dataoperation="AND">
                <trustee_domain>NT AUTHORITY</trustee_domain>
                <trustee_sid>S-1-5-18</trustee_sid>
                <standard_deletedatatype="string">AUDIT_SUCCESS</standard_delete>
                <standard_read_controldatatype="string">AUDIT_FAILURE</standard_read_contr
                <file_read_attributesdatatype="string">AUDIT_SUCCESS_FAILURE</file_read_a
                <file_write_attributesdatatype="string">AUDIT_NONE</file_write_attributes>
            </data>
        </fileauditedpermissions_test>
```

object section

<path>

Specifies the absolute path to a file on the machine. This path can be created from multiple components that are added together. When a pattern match operator is used, the corresponding regular expression is matched against the set of absolute path strings. These string would not include the '.' and '..' notations. This means that a '.*' component of a regular expression will not only match all files in the specified directories, but all subdirectories, their subdirectories, etc.

Parent Test:	File Audited Permissions Test
Cardinality:	1
Content:	none
Valid Datatypes:	component
Valid Operators:	equals, not equal, pattern match

<trustee_name>

This element specifies the trustee name associated with a particular SACL. A trustee can be a user, group, or program (such as a Windows service)

Parent Test:	File Audited Permissions Test
Cardinality:	1
Content:	string

Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<trustee_domain>

The domain of the specified trustee name.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<trustee_sid>

The security identifier (SID) of the specified trustee name.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<standard_delete>

The right to delete the object.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_read_control>

The right to read the information in the object's security descriptor, not including the information in the SACL.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_write_dac>

The right to modify the DACL in the object's security descriptor.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_write_owner>

The right to change the owner in the object's security descriptor.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_synchronize>

Windows NT/2000: The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<access_system_security>

Indicates access to a system access control list (SACL).

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<generic_read>

Read access.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<generic_write>

Write access.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<generic_execute>

Execute access.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string

Valid Operators:	equals, not equal

<generic_all>

Read, write, and execute access.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_read_data>

Grants the right to read data from the file

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_write_data>

Grants the right to write data to the file.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_append_data>

Grants the right to append data to the file.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string

Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_read_ea>

Grants the right to read extended attributes.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_write_ea>

Grants the right to write extended attributes.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_execute>

Grants the right to execute a file.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_delete_child>

Right to delete a directory and all the files it contains (its children), even if the files are read-only.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1

Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_read_attributes>

Grants the right to read file attributes.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<file_write_attributes>

Grants the right to change file attributes.

Parent Test:	File Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

File Effective Rights Test

<fileeffectiverights_test>

This test looks at the effective rights of a given file that a discretionary access control list (DACL) structure grants to a specified trustee. The trustee's effective rights are determined by checking all access-allowed and access-denied access control entries (ACEs) in the DACL. Note that the rights expressed in this test correspond to the diffent bits allocated to access mask for a file. This means that certain rights that represent combinations of other rights are not expressed. For example STANDARD_RIGHTS_ALL and FILE_ALL_ACCESS. For help with this test see the GetEffectiveRightsFromAcl() api.

Extends:	standardTestType

Valid Sections:	notes, object, data

```
<fileeffectiverights_testid="wet-0"check="at least one"comment="a file exists with the specified
    <oval:notes>
        <oval:note>This is an example test written under version 4 of the OVAL schema. It
        short overview of the test and might not contain every possible child element.</oval
    </oval:notes>
    <object>
        <path>
            <componenttype="registry_value">HKEY_LOCAL_MACHINE\SOFTWAR
            NT\CurrentVersion\SystemRoot</component>
            <componenttype="literal">\system32\mshtml.dll</component>
        </path>
        <trustee_name>SYSTEM</trustee_name>
    </object>
    <dataoperation="AND">
        <trustee_domain>NT AUTHORITY</trustee_domain>
        <trustee_sid>S-1-5-18</trustee_sid>
        <standard_deletedatatype="boolean">0</standard_delete>
        <standard_read_controldatatype="boolean">1</standard_read_control>
        <file_read_attributesdatatype="boolean">true</file_read_attributes>
        <file_write_attributesdatatype="boolean">false</file_write_attributes>
    </data>
</fileeffectiverights_test>
```

object section

<path>

Specifies the absolute path to a file on the machine. This path can be created from multiple components that are added together. When a pattern match operator is used, the corresponding regular expression is matched against the set of absolute path strings. These string would not include the '.' and '..' notations. This means that a '.*' component of a regular expression will not only match all files in the specified directories, but all subdirectories, their subdirectories, etc.

Parent Test:	File Effective Rights Test
Cardinality:	1
Content:	none
Valid Datatypes:	component
Valid Operators:	equals, not equal, pattern match

<trustee_name>

This element specifies the trustee name associated with a particular DACL. A trustee can be a user, group, or program (such as a Windows service)

Parent Test:	File Effective Rights Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<trustee_domain>

The domain of the specified trustee name.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<trustee_sid>

The security identifier (SID) of the specified trustee name.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<standard_delete>

The right to delete the object.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_read_control>

The right to read the information in the object's security descriptor, not including the information in the SACL.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_write_dac>

The right to modify the DACL in the object's security descriptor.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_write_owner>

The right to change the owner in the object's security descriptor.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_synchronize>

Windows NT/2000: The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<access_system_security>

Indicates access to a system access control list (SACL).

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_read>

Read access.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_write>

Write access.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_execute>

Execute access.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_all>

Read, write, and execute access.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_read_data>

Grants the right to read data from the file

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_write_data>

Grants the right to write data to the file.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean

Valid Operators:	equals, not equal

<file_append_data>

Grants the right to append data to the file.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_read_ea>

Grants the right to read extended attributes.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_write_ea>

Grants the right to write extended attributes.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_execute>

Grants the right to execute a file.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean

Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_delete_child>

Right to delete a directory and all the files it contains (its children), even if the files are read-only.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_read_attributes>

Grants the right to read file attributes.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_write_attributes>

Grants the right to change file attributes.

Parent Test:	File Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

Group Test

<group_test>

The windows group test allows the different users that belong to specific groups be tested. Note that the user element can appear an unlimited number of times. In such cases, the test is wether the specified user belongs to ALL the included groups (data operator is AND) or that the user belongs to one of the included groups (data operator is OR).

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<group_testid="wgt-0"check="all"comment="dave and jon are members of the
Administrators group">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <group>Administrators</group>
        </object>
        <dataoperation="AND">
                <enabled>true</enabled>
                <user>dave</user>
                <user>jon</user>
        </data>
</group_test>
```

object section

<group>

A string that represents the name of a particular group.

Parent Test:	Group Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<enabled>

This element holds a boolean value that specifies whether the particular group is enabled or not.

Parent Test:	Group Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<user>

A string that represents the name of a particular user. This element can be included multiple times in order to test that a group contains a number of different users.

Parent Test:	Group Test
Cardinality:	0-n
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

Interface Test

<interface_test>

Enumerate various attributes about the interfaces on a system. Each interface is uniquely identified by either its name or an index number. For help with this test see the MIB_IFROW and MIB_IPADDRROW structures.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<interface_testid="wit-0"check="all"comment="the interface exists with the specified
properties">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <name>Intel(R) PRO/1000 MTW Network Connection - Packet Scheduler
                Miniport</name>
        </object>
        <dataoperation="AND">
                <type>MIB_IF_TYPE_ETHERNET</type>
                <hardware_addr>33-22-11-AA-CC-BB</hardware_addr>
                <inet_addr>123.45.67.89</inet_addr>
        </data>
</interface_test>
```

object section

<name>

This element specifies the name of an interface.

Parent Test:	Interface Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<index>

This element specifies index that identifies the interface.

Parent Test:	Interface Test

Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<type>

This element specifies the type of interface which is limited to certain set of values.

Parent Test:	Interface Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<hardware_addr>

This element specifies the the physical address of the adapter for this interface.

Parent Test:	Interface Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<inet_addr>

This element specifies the IP address.

Parent Test:	Interface Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<broadcast_addr>

This element specifies the broadcast address. A broadcast address is typically the IP address with the host

portion set to either all zeros or all ones.

Parent Test:	Interface Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<netmask>

This element specifies the subnet mask for the IP address.

Parent Test:	Interface Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<addr_type>

This element specifies the address type or state of a specific interface. Each interface can be associated with more than one value meaning the addr_type element can occur multiple times.

Parent Test:	Interface Test
Cardinality:	0-n
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

Lockout Policy Test

<lockoutpolicy_test>

The lockoutpolicy test enumerates various attributes associated with lockout information for users and global groups in the security database.

Extends:	standardTestType
Valid Sections:	notes, data

```
<lockoutpolicy_testid="wlt-0"check="all"comment="specific lockout policies are set">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <dataoperation="AND">
                <force_logoffdatatype="int">60</force_logoff>
                <lockout_durationdatatype="int">30</lockout_duration>
                <lockout_observation_windowdatatype="int">5</lockout_observation_window>
                <lockout_thresholddatatype="int">3</lockout_threshold>
        </data>
</lockoutpolicy_test>
```

data section

<force_logoff>

Specifies, in seconds, the amount of time between the end of the valid logon time and the time when the user is forced to log off the network. A value of TIMEQ_FOREVER indicates that the user is never forced to log off. A value of zero indicates that the user will be forced to log off immediately when the valid logon time expires. See the USER_MODALS_INFO_0 structure returned by a call to NetUserModalsGet().

Parent Test:	Lockout Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<lockout_duration>

Specifies, in seconds, how long a locked account remains locked before it is automatically unlocked. See the USER_MODALS_INFO_3 structure returned by a call to NetUserModalsGet().

Parent Test:	Lockout Policy Test

Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<lockout_observation_window>

Specifies the maximum time, in seconds, that can elapse between any two failed logon attempts before lockout occurs. See the USER_MODALS_INFO_3 structure returned by a call to NetUserModalsGet().

Parent Test:	Lockout Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<lockout_threshold>

Specifies the number of invalid password authentications that can occur before an account is marked "locked out." See the USER_MODALS_INFO_3 structure returned by a call to NetUserModalsGet().

Parent Test:	Lockout Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

---

Metabase Test

<metabase_test>

This test gathers information from the specified metabase keys.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<metabase_testid="wmt-0"check="all"comment="HTTP is enabled">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <keyoperator="pattern match">^LM\\W3SVC\\.*$</key>
                <iddatatype="int">1016</id>
        </object>
        <data>
                <datadatatype="int"operator="not equal">4</data>
        </data>
</metabase_test>
```

object section

<key>

This element describes a metabase key to be tested.

Parent Test:	Metabase Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<id>

The id element specifies a particular object under the metabase key. If nillable is set to true, then the id element should be ignored during analysis.

Parent Test:	Metabase Test
Cardinality:	1
Content:	integer

Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

data section

<name>

This element describes the name of the specified metabase object.

Parent Test:	Metabase Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<user_type>

A DWORD that specifies the user type of the data. See the METADATA_RECORD structure.

Parent Test:	Metabase Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<data_type>

Identifies the type of data in the metabase entry. See the METADATA_RECORD structure.

Parent Test:	Metabase Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<data>

The actual data of the named item under the specified metabase key

Parent Test:	Metabase Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	binary, boolean, float, int, string
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal, bitwise and, bitwise or, pattern match

---

Password Policy Test

<passwordpolicy_test>

Test specific policy associated with passwords. Information is stored in the SAM or Active Directory but is encrypted or hidden so the registry_test and activedirectory_test are of no use. If this can be figured out, then the password_policy test is not needed.

Extends:	standardTestType
Valid Sections:	notes, data

```
<passwordpolicy_testid="wdt-0"check="all"comment="specific password policies are set">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <dataoperation="AND">
                <max_passwd_agedatatype="int">-1</max_passwd_age>
                <min_passwd_agedatatype="int">3600</min_passwd_age>
                <min_passwd_lendatatype="int">8</min_passwd_len>
                <password_hist_lendatatype="int">5</password_hist_len>
        </data>
</passwordpolicy_test>
```

data section

<max_passwd_age>

Specifies, in seconds, the maximum allowable password age. A value of TIMEQ_FOREVER (-1) indicates that the password never expires. The minimum valid value for this element is ONE_DAY (86400).

Parent Test:	Password Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<min_passwd_age>

Specifies the minimum number of seconds that can elapse between the time a password changes and when it can be changed again. A value of zero indicates that no delay is required between password updates.

Parent Test:	Password Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<min_passwd_len>

Specifies the minimum allowable password length. Valid values for this element are zero through PWLEN.

Parent Test:	Password Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<password_hist_len>

Specifies the length of password history maintained. A new password cannot match any of the previous usrmod0_password_hist_len passwords. Valid values for this element are zero through DEF_MAX_PWHIST.

Parent Test:	Password Policy Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<password_complexity>

A boolean value that signifies whether passwords must meet the complexity requirements put forth by the operating system.

Parent Test:	Password Policy Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<reversible_encryption>

Determines whether Windows 2000 Server, Windows 2000 Professional, and Windows XP Professional store passwords using reversible encryption.

Parent Test:	Password Policy Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

Port Test

<port_test>

Information about listening ports ports.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<port_testid="wqt-0"check="all"comment="TCP port 443 is open for listening by the
process with an id of 3796">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <local_addressoperator="pattern match">^.*$</local_address>
                <local_portdatatype="int">443</local_port>
                <protocol>TCP</protocol>
        </object>
        <data>
                <piddatatype="int">3796</pid>
        </data>
</port_test>
```

object section

<local_address>

This element specifies the local IP address the listening port is bound to.

Parent Test:	Port Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<local_port>

This element specifies the number assigned to the local listening port.

Parent Test:	Port Test

Cardinality:	1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<protocol>

This element specifies the type of listening port. It is restricted to either TCP or UDP.

Parent Test:	Port Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

data section

<pid>

The id given to the process that is associated with the specified listening port.

Parent Test:	Port Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

---

Process Test

<process_test>

Information about running processes.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<process_testid="wct-0"check="all"comment="there exists an inetinfo process with a pid of
1680">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <command_line>C:\WINDOWS\System32
                \inetsrv\inetinfo.exe</command_line>
        </object>
        <data>
                <piddatatype="int">1680</pid>
        </data>
</process_test>
```

object section

<command_line>

The command line used to start the process.

Parent Test:	Process Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<pid>

The id given to the process that is created for a specified command line.

Parent Test:	Process Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<ppid>

The id given to the parent of the process that is created for the specified command line

Parent Test:	Process Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<priority>

The base priority of the process

Parent Test:	Process Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<image_path>

This field contains the DOS Path of the image file.

Parent Test:	Process Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<current_dir>

This field has the current path in DOS format ("C:\WINDOWS")

Parent Test:	Process Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

Registry Test

<registry_test>

The windows registry test specifies a particular registry key (or keys) to test

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<registry_testid="wrt-0"check="all"comment="Windows XP is installed">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <hive>HKEY_LOCAL_MACHINE</hive>
                <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
                <name>CurrentVersion</name>
        </object>
        <dataoperation="AND">
                <valueoperator="equals">5.1</value>
        </data>
</registry_test>
```

object section

<hive>

The hive that the registry key belongs to.

Parent Test:	Registry Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key>

This element describes a registry key to be tested. Note that the hive portion of the string should not be inclueded, as this data should be found under the hive element.

Parent Test:	Registry Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<name>

This element describes the name of a value of a registry key. If the nillable attribute is set to true, then the name element should not be used in analysis.

Parent Test:	Registry Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<type>

Specifies the type of data stored by the registry key.

Parent Test:	Registry Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<value>

The actual value of the specified registry key.

Parent Test:	Registry Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	binary, boolean, float, int, string
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal, bitwise and, bitwise or, pattern match

---

Regkey Audited Permissions Test

<regkeyauditedpermissions_test>

This test looks at the audited access rights of a given registry key that a system access control list (SACL) structure grants to a specified trustee. The trustee's audited access rights are determined checking all access control entries (ACEs) in the SACL. For help with this test see the GetAuditedPermissionsFromAcl() api.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<regkeyauditedpermissions_testid="wyt-0"check="at least one"comment="a registry key exists v
the specified audit rights">
    <oval:notes>
```

```
            <oval:note>This is an example test written under version 4 of the OVAL schema. It
            meant to give a short overview of the test and might not contain every possible child
            element.</oval:note>
        </oval:notes>
        <object>
            <hive>HKEY_LOCAL_MACHINE</hive>
            <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
            <trustee_name>SYSTEM</trustee_name>
        </object>
        <dataoperation="AND">
            <trustee_domain>NT AUTHORITY</trustee_domain>
            <trustee_sid>S-1-5-18</trustee_sid>
            <standard_deletedatatype="string">AUDIT_SUCCESS</standard_delete>
            <standard_read_controldatatype="string">AUDIT_FAILURE</standard_read_contr
            <key_query_valuedatatype="string">AUDIT_SUCCESS_FAILURE</key_query_v
            <key_set_valuedatatype="string">AUDIT_NONE</key_set_value>
        </data>
    </regkeyauditedpermissions_test>
```

object section

<hive>

This element specifies the hive of a registry key on the machine from which to retrieve the SACL.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key>

This element specifies a registry key on the machine from which to retrieve the SACL. Note that the hive portion of the string should not be inclueded, as this data should be found under the hive element.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	1
Content:	string
Valid Datatypes:	string

Valid Operators:	equals, not equal, pattern match

<trustee_name>

This element specifies the trustee name associated with a particular SACL. A trustee can be a user, group, or program (such as a Windows service)

Parent Test:	Regkey Audited Permissions Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<trustee_domain>

The domain of the specified trustee name.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<trustee_sid>

The security identifier (SID) of the specified trustee name.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<standard_delete>

The right to delete the object.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_read_control>

The right to read the information in the object's security descriptor, not including the information in the SACL.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_write_dac>

The right to modify the DACL in the object's security descriptor.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_write_owner>

The right to change the owner in the object's security descriptor.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<standard_synchronize>

Windows NT/2000: The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<access_system_security>

Indicates access to a system access control list (SACL).

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<generic_read>

Read access.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<generic_write>

Write access.

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string

Valid Operators:	equals, not equal

<generic_execute>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<generic_all>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_query_value>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_set_value>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_create_sub_key>

Parent Test:	Regkey Audited Permissions Test

Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_enumerate_sub_keys>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_notify>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_create_link>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_wow64_64key>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_wow64_32key>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key_wow64_res>

Parent Test:	Regkey Audited Permissions Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

Regkey Effective Rights Test

<regkeyeffectiverights_test>

This test looks at the effective rights of a given registry key that a discretionary access control list (DACL) structure grants to a specified trustee. The trustee's effective rights are determined checking all access-allowed and access-denied access control entries (ACEs) in the DACL. Note that the rights expressed in this test correspond to the diffent bits allocated to access mask for a registry key. This means that certain rights that represent combinations of other rights are not expressed. For example STANDARD_RIGHTS_ALL and KEY_ALL_ACCESS. For help with this test see the GetEffectiveRightsFromAcl() api.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<regkeyeffectiverights_testid="wzt-0"check="at least one"comment="a registry key exists
with the specified rights">
        <oval:notes>
```

```
            <oval:note>This is an example test written under version 4 of the OVAL
            schema. It is meant to give a short overview of the test and might not contain
            every possible child element.</oval:note>
        </oval:notes>
        <object>
            <hive>HKEY_LOCAL_MACHINE</hive>
            <key>SOFTWARE\Microsoft\Windows NT\CurrentVersion</key>
            <trustee_name>SYSTEM</trustee_name>
        </object>
        <dataoperation="AND">
            <trustee_domain>NT AUTHORITY</trustee_domain>
            <trustee_sid>S-1-5-18</trustee_sid>
            <standard_deletedatatype="boolean">0</standard_delete>
            <standard_read_controldatatype="boolean">1</standard_read_control>
            <key_query_valuedatatype="boolean">true</key_query_value>
            <key_set_valuedatatype="boolean">false</key_set_value>
        </data>
    </regkeyeffectiverights_test>
```

object section

<hive>

This element specifies the hive of a registry key on the machine from which to retrieve the DACL.

Parent Test:	Regkey Effective Rights Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal

<key>

This element specifies a registry key on the machine from which to retrieve the DACL. Note that the hive portion of the string should not be inclueded, as this data should be found under the hive element.

Parent Test:	Regkey Effective Rights Test
Cardinality:	1
Content:	string
Valid Datatypes:	string

Valid Operators:	equals, not equal, pattern match

<trustee_name>

This element specifies the trustee name associated with a particular DACL. A trustee can be a user, group, or program (such as a Windows service)

Parent Test:	Regkey Effective Rights Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<trustee_domain>

The domain of the specified trustee name.

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<trustee_sid>

The security identifier (SID) of the specified trustee name.

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<standard_delete>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_read_control>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_write_dac>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_write_owner>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<standard_synchronize>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean

Valid Operators:	equals, not equal

<access_system_security>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_read>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_write>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_execute>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<generic_all>

Parent Test:	Regkey Effective Rights Test

Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_query_value>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_set_value>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_create_sub_key>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_enumerate_sub_keys>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_notify>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_create_link>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_wow64_64key>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_wow64_32key>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<key_wow64_res>

Parent Test:	Regkey Effective Rights Test
Cardinality:	0-1

Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

Text File Content Test

<textfilecontent_test>

This test has been deprecated in version 4.1 of the windows-schema and will be removed completely in version 5. It is recommended that all future OVAL Content use the textfilecontent_test found in the independent-schema.

The textfilecontent test looks at the contents of a text file (aka a configuration file) by looking at individual lines.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<textfilecontent_testid="wtt-0"check="all"comment="the enable parameter in helpctr.txt is set to
    <oval:notes>
        <oval:note>This is an example test written under version 4 of the OVAL schema. It
        short overview of the test and might not contain every possible child element.</oval
    </oval:notes>
    <object>
        <path>
            <componenttype="registry_value">HKEY_LOCAL_MACHINE\SOFTWAR
            NT\CurrentVersion\SystemRoot</component>
            <componenttype="literal">\system32\helpctr.txt</component>
        </path>
        <lineoperator="pattern match">enable = (true|false)</line>
    </object>
    <dataoperation="AND">
        <subexpressionoperator="equals">true</subexpression>
    </data>
</textfilecontent_test>
```

object section

<path>

Specifies the absolute path to a file on the machine. This path can be created from multiple components that are added together. When a pattern match operator is used, the corresponding regular expression is matched against the set of absolute path strings. These string would not include the '.' and '..' notations. This means that a '.*' component of a regular expression will not only match all files in the specified directories, but all subdirectories, their subdirectories, etc.

Parent Test:	Text File Content Test
Cardinality:	1
Content:	none
Valid Datatypes:	component
Valid Operators:	equals, not equal, pattern match

<line>

The line element represents a line in the file and is represented using a regular expression.

Parent Test:	Text File Content Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	pattern match

data section

<subexpression>

Each subexpression in the regular expression of the line element is then tested against the value specified in the subexpression element.

Parent Test:	Text File Content Test
Cardinality:	0-n
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

User Test

<user_test>

The windows user test allows the different groups that a user belongs to be tested. Note that the group element can appear an unlimited number of times. In such cases, the test is wether the specified group contains ALL the included users (data operator is AND) or that the group contains at least one of the included users (data operator is OR).

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<user_testid="wut-0"check="all"comment="drew is member of the Administrators group">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <user>drew</user>
        </object>
        <data>
                <enabled>true</enabled>
                <group>Administrators</group>
        </data>
</user_test>
```

object section

<user>

A string that represents the name of a particular user.

Parent Test:	User Test
Cardinality:	1

Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<enabled>

This element holds a boolean value that specifies whether the particular user account is enabled or not.

Parent Test:	User Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<group>

A string the represents the name of a particular group. This element can be included multiple times in order to test that a user is a member of a number of different groups.

Parent Test:	User Test
Cardinality:	0-n
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

Volume Test

<volume_test>

The volume test enumerates various attributes about a particular volume mounted to a machine. This includes the various system flags returned by GetVolumeInformation().

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<volume_testid="wvt-0"check="all"comment="the OVAL volume supports named
streams">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <rootpath>
                        <componenttype="literal">\\MyServer\MyShare\</component>
                </rootpath>
        </object>
        <dataoperation="AND">
                <file_system>NTFS</file_system>
                <name>OVAL</name>
                <file_named_streamsdatatype="boolean">true</file_named_streams>
        </data>
</volume_test>
```

object section

<rootpath>

A string that contains the root directory of the volume to be described. A trailing backslash is required. For example, you would specify \\MyServer\MyShare as "\\MyServer\MyShare\", or the C drive as "C:\".

Parent Test:	Volume Test
Cardinality:	1
Content:	none
Valid Datatypes:	component
Valid Operators:	equals, not equal, pattern match

data section

<file_system>

The type of filesystem. For example FAT or NTFS.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<name>

The name of the volume.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<serial_number>

The volume serial number.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	integer
Valid Datatypes:	integer
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal

<file_named_streams>

The file system supports named streams.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean

Valid Operators:	equals, not equal

<file_read_only_volume>

The specified volume is read-only.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_supports_object_ids>

The file system supports object identifiers.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_supports_reparse_points>

The file system supports reparse points.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_supports_sparse_files>

The file system supports sparse files.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean

Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<file_volume_quotas>

The file system supports disk quotas.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_case_is_preserved>

The file system preserves the case of file names when it places a name on disk.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_case_sensitive>

The file system supports case-sensitive file names.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_file_compression>

The file system supports file-based compression.

Parent Test:	Volume Test
Cardinality:	0-1

Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_file_encryption>

The file system supports the Encrypted File System (EFS).

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_persistent_acls>

The file system preserves and enforces ACLs. For example, NTFS preserves and enforces ACLs, and FAT does not.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_unicode_stored_on_disk>

The file system supports Unicode in file names as they appear on disk.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

<fs_vol_is_compressed>

The specified volume is a compressed volume; for example, a DoubleSpace volume.

Parent Test:	Volume Test
Cardinality:	0-1
Content:	boolean
Valid Datatypes:	boolean
Valid Operators:	equals, not equal

WMI Test

<wmi_test>

The wmi_test outlines information to be checked through Microsoft's WMI interface. WMI is a layer on top of the actual data and many times, the information being collected can also be retrieved using a registry test, active directory test, etc. It is recommended that the lowest level approach to data collection and analysis be taken to avoid any possible corruption that might exist on the machine.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<wmi_testid="wwt-0"check="at least one"comment="correct permission is assigned">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <namespaceoperator="pattern
                match">^root\\sms\\site_.*\\SMS_SiteControlItem$</namespace>
                <wql>SELECT SMS_Query.name FROM
                SMS_UserInstancePermissionNames, SMS_Query WHERE
                SMS_UserInstancePermissionNames.instancekey = SMS_query.queryid AND
                SMS_UserInstancePermissionNames.objectkey = 7</wql>
        </object>
        <data>
                <resultdatatype="string"operator="equals">Fred</result>
        </data>
</wmi_test>
```

object section

<namespace>

Specifies which WMI namespace to look under. Each WMI provider normally registers its own WMI namespace and then all its classes within that namespace. For example, all Win32 WMI classes can be found in the namespace "root\cimv2", all IIS WMI classes can be found at "root\microsoftiisv2", and all LDAP WMI classes can be found at "root\directory\ldap".

Parent Test:	WMI Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<wql>

A WQL query used to identify the object(s) to test against. Any valid WQL query is usable with one exception, at most one field is allowed in the SELECT portion of the query. For example SELECT name FROM ... is valid, as is SELECT 'true' FROM ..., but SELECT name, number FROM ... is not valid. This is because the result element in the data section is only designed to work against a single field.

Parent Test:	WMI Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<result>

The result element specifies how to test objects in the result set of the specified WQL statement. Only one comparable field is allowed. So if the WQL statement look like 'SELECT name FROM ...', then a result element with a value of 'Fred' would test that value against the names returned by the WQL statement.

Parent Test:	WMI Test
Cardinality:	0-1
Content:	string

Valid Datatypes:	binary, boolean, float, int, string
Valid Operators:	equals, not equal, greater than, less than, greater than or equal, less than or equal, bitwise and, bitwise or, pattern match

XML File Content Test

<xmlfilecontent_test>

This test has been deprecated in version 4.1 of the windows-schema and will be removed completely in version 5. It is recommended that all future OVAL Content use the xmlfilecontent_test found in the independent-schema.

The xmlfilecontent test uses Xpath to explore the contents of an xml file. The value element checks the value of the nodes found.

Extends:	standardTestType
Valid Sections:	notes, object, data

```
<xmlfilecontent_testid="wxt-0"check="none exist"comment="there does not exists an
Andrew object in fred.xml">
        <oval:notes>
                <oval:note>This is an example test written under version 4 of the OVAL
                schema. It is meant to give a short overview of the test and might not contain
                every possible child element.</oval:note>
        </oval:notes>
        <object>
                <path>
                        <componenttype="literal">c:\fred.xml</component>
                </path>
                <xpath>/people/name</xpath>
        </object>
        <dataoperation="AND">
                <value_ofoperator="equals">Andrew</value_of>
        </data>
</xmlfilecontent_test>
```

object section

<path>

Specifies the absolute path to a file on the machine. This path can be created from multiple components that are added together. When a pattern match operator is used, the corresponding regular expression is matched against the set of absolute path strings. These string would not include the '.' and '..' notations. This means that a '.*' component of a regular expression will not only match all files in the specified directories, but all subdirectories, their subdirectories, etc.

Parent Test:	XML File Content Test
Cardinality:	1
Content:	none
Valid Datatypes:	component
Valid Operators:	equals, not equal, pattern match

<xpath>

Specifies an Xpath expression describing the nodes to look at.

Parent Test:	XML File Content Test
Cardinality:	1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

data section

<value_of>

The value element checks the value of the nodes found.

Parent Test:	XML File Content Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string
Valid Operators:	equals, not equal, pattern match

<platform>

The valid platforms for the Microsoft Windows Family.

-- Microsoft Windows 95
-- Microsoft Windows 98
-- Microsoft Windows ME
-- Microsoft Windows NT
-- Microsoft Windows 2000
-- Microsoft Windows XP
-- Microsoft Windows Server 2003

Complex Types

This section describes any global complex types defined in the schema. These types can be instantiated by elements in this schema as well as elements in other schemas. Note that in the tables outlining possible attributes and child elements, square brackets [] means that the item is optional.

-- subtestAuditType --

The subtestAuditType restricts a string value to a specific set of values: AUDIT_NONE, AUDIT_SUCCESS, AUDIT_FAILURE, and AUDIT_SUCCESS_FAILURE. These values describe which audit records should be generated.

Extends:	oval:subtestStringType
Attributes:	(includes oval:subtestAttributes)
Content:	string
Child Elements:	none

-- componentType --

The componentType allows a value to be obtained by combining pieces from different sources. Each string defined by the different component elements is concatenated together to form the final string used. Each child component element has an attribute called type. The value of this attribute determines where to get the string used to build the file path. A type of literal means to use the value of the child component element as is, and to just concatenated it to the other strings. If a pattern match operator has been specified with a componentType, then the final string should be thought of as the pattern to test. As of Version 4 of the OVAL schema, pattern match can not be specified for the idividual components.

Extends:	oval:subtestBaseType
Attributes:	(includes oval:subtestAttributes)
Content:	none
Child Elements:	component

-- subtestFileVersionType --

The subtestFileVersionType allows the different portions of a windows file version to be represented. A windows file version is made up of for distinct parts: a major version, a minor version, a build number, and a private number. Each part is an integer

Extends:	oval:subtestBaseType
Attributes:	(includes oval:subtestAttributes)
Content:	none
Child Elements:	major, minor, build, private

-- subtestHiveType --

The subtestHiveType restricts a string value to a specific set of values: HKEY_CLASSES_ROOT, HKEY_CURRENT_CONFIG, HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, and HKEY_USERS. These values describe the possible hives in the registry.

Extends:	oval:subtestStringType
Attributes:	(includes oval:subtestAttributes)
Content:	string
Child Elements:	none