

# - Open Vulnerability and Assessment Language - Element Dictionary

- Schema: HP-UX Definition
- Version: 4.2
- Release Date: 2 December 2005

The following is a description of the elements, types, and attributes that compose the HP-UX 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 HP-UX Definition Schema was initially developed by ThreatGuard. Many thanks to their contributions to OVAL and the security community.

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.

---

## Getconf Test

---

### <getconf\_test>

From /usr/bin/getconf. See getconf manpage for specific fields

Extends:	standardTestType
Valid Sections:	object, data



```
<getconf_testid="hgt-0"check="all"comment="The kernel is a 64-bit installation.">
  <oval:notes>
    <oval:note>This is an example test. It is meant to give a short overview of the
      test and might not contain every possible child element.</oval:note>
  </oval:notes>
  <object>
    <parameter_name>KERNEL_BITS</parameter_name>
  </object>
  <dataoperation="AND">
    <outputoperator="equals">64</output>
  </data>
</getconf_test>
```

## object section

---

---

### <parameter\_name>

This is the parameter name to check

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

### <pathname>

This is the pathname to check

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

## data section

---

---

## <output>

The output produced by the getconf command.

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

---

---

## Inetd Test

---

## <inetd\_test>

See manpage for inetd.conf for column definitions

Extends:	standardTestType
Valid Sections:	notes, object, data

---

---

## object section

---

---

## <server\_program>

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

## data section

---

---

**<socket\_type>**

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

**<exec\_as\_user>**

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

**<protocol>**

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

**<service\_name>**

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

**<server\_program\_arguments>**

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

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

### <wait\_status>

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

---



---

## Patch Test

---

### <patch\_test>

From /usr/sbin/swlist -l patch PHxx\_yyyyy. See manpage for specific fields

Extends:	standardTestType
Valid Sections:	object, data

---



---

## object section

---



---

### <area\_patched>

The first four characters in HP-UX patch names indicate the area of software being patched. PHCO = General HP-UX commands. PHKL = Kernel patches. PHNE = Network specific patches. PHSS = All other subsystems (X11, starbase, etc.).

Parent Test:	Patch Test
Cardinality:	1
Content:	string
Valid Datatypes:	string

Valid Operators:	equals, not equal, pattern match
------------------	----------------------------------

## <patch\_number>

The sixth through tenth characters in HP-UX patch names represent a unique numeric identifier for the patch.

Parent Test:	Patch 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

## Password Test

## <password\_test>

/etc/passwd. See passwd(4).

Extends:	standardTestType
Valid Sections:	notes, object, data

## object section

## <username>

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

## data section

---

---

### <password>

This is the encrypted version of the user's password.

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

### <user\_id>

The numeric user id, or uid, is the third column of each user's entry in /etc/passwd. This element represents the owner of the file.

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

### <group\_id>

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

### <gcos>

The GCOS field has no defined syntax and is generally used for personal information about the user - full name, phone number, room number, etc.

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

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

### <home\_dir>

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

### <login\_shell>

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

---



---

## Shadow Test

---

### <shadow\_test>

This test allows you to check information from the /etc/shadow file for a specific user. This file contains a user's password, but also their password aging and lockout information.

Background: Unix systems are generally configured to only allow a given password to last for a fixed period of time. When this time, the chg\_req parameter, is near running out, the system begins warning the user at each login. How soon before the expiration the user receives these warnings is specified in exp\_warn. The only hiccup in this design is that a user may not login in time to ever receive a warning before account expiration. The exp\_inact parameter gives the sysadmin flexibility so that a user who reaches the end of their expiration time gains exp\_inact more days to login and change their password manually.

Extends:	standardTestType
Valid Sections:	notes, object, data



---

---

## object section

---

---

### <username>

This is the name of the user being checked.

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

## data section

---

---

### <password>

This is the encrypted version of the user's password.

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

### <chg\_lst>

This is the date of the last password change in days since 1/1/1970.

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

### <chg\_allow>

This specifies the minimum period in days that must expire before the password can be changed.

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

### <chg\_req>

This describes how long a user can keep a password before the system forces user to change it.

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

### <exp\_warn>

The number of days the user is warned before his password expires.

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

### <exp\_inact>

This describes how many days of account inactivity the system will wait after a password expires before locking the account.

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

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

### <exp\_date>

This specifies in days since 1/1/1970 when the account's password will expire.

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

### <flag>

This is a reserved field that the shadow file may use in the future.

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

---



---

## swlist Test

---

### <swlist\_test>

Output of /usr/sbin/swlist command. Note: A quick way to check for the installation of a specific fileset is to use the command 'swlist -a version -l fileset filesetname'. See manpage for swlist for explanation of additional command options.

Extends:	standardTestType
Valid Sections:	object, data

---

---

## object section

---

---

### <swlist>

This is the name of the bundle or fileset to check.

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

## data section

---

---

### <bundle>

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

### <fileset>

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

### <version>

Parent Test:	swlist Test
--------------	-------------

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

**<title>**

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

**<vendor>**

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

---



---

## Trusted Test

---

**<trusted\_test>**

This test allows for analysis of account settings in trusted HP-UX installations

Extends:	standardTestType
Valid Sections:	object, data

---

## object section

---



---

### **<username>**

This is the name of the user being checked

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

## **data section**

---

---

### **<uid>**

The user's ID

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

### **<password>**

This is the encrypted version of the user's password

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

### **<account\_owner>**

The Account owner for pseudo-users

Parent Test:	Trusted Test

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

### <boot\_auth>

Boot authorization

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

### <audit\_id>

getprpwaid uses the audit ID rather than the UID

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

### <audit\_flag>

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

### <pw\_change\_min>

Minimum time between password changes

Parent Test:	Trusted Test

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

### <pw\_max\_size>

Maximum password length in characters

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

### <pw\_expiration>

Password expiration time in seconds

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

### <pw\_life>

Trusted lifetime, after which the account is locked

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

### <pw\_change\_s>

Time of last successful password change

--	--



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

### <pw\_change\_u>

Time of last unsuccessful password change

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

### <acct\_expire>

Absolute account lifetime in seconds

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

### <max\_llogin>

Maximum time allowed between logins before the account is locked

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

### <exp\_warning>

The time in seconds before expiration when a warning will appear

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

**<usr\_chg\_pw>**

Who can change this user's password

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

**<gen\_pw>**

Allows user to use system-generated passwords

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

**<pw\_restrict>**

Whether a triviality check is performed on user-generated passwords

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

### <pw\_null>

Determines if null passwords are allowed for this account

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

### <pw\_gen\_char>

Allows password generator to use random printable ASCII characters

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

### <pw\_gen\_let>

Allows password generator to use random letters

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

### <login\_time>

Specifies the times when the user may login to this account

Parent Test:	Trusted Test
Cardinality:	0-1
Content:	string
Valid Datatypes:	string

Valid Operators:	equals, not equal, pattern match
------------------	----------------------------------

### <pw\_changer>

The user ID of the user who last changed the password on the user's account, if it was not the account owner

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

### <login\_time\_s>

The time of the last successful login using this account

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

### <login\_time\_u>

The time of the last unsuccessful login using this account

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

### <login\_tty\_s>

The terminal or remote host associated with the last successful login to the account

Parent Test:	Trusted Test
Cardinality:	0-1

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

### <login\_tty\_u>

The terminal or remote hosts associated with the last unsuccessful login to the account

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

### <num\_u\_logins>

The number of unsuccessful login attempts since that last successful login

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

### <max\_u\_logins>

The maximum number of unsuccessful login attempts before the account is locked

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

### <lock\_flag>

Indicates whether the administrative lock on the account is set

Parent Test:	Trusted Test
--------------	--------------

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

---



---

### <platform>

The valid platforms for the Hewlett-Packard HP-UX family.

- HP-UX 10
- HP-UX 11
- HP-UX 11i