V2Check

Manual

(version 3.0)

Public Record Office Victoria

Further copies of this document can be obtained from the PROV Web site
http://www.prov.vic.gov.au/

Version	Version Date	Details
1.0	22 May 2006	Released
2.0	30 June 2015	Add virus checking etc
3.0	18 March 2020	Updated

Contents

1. Purpose of Document

This document describes how to run the tool that will test VEOs for compliance with PROS 99/007 (Version 1 and 2).

2. Overview of running VEO Check II

To test VEOs, the user will

- Install the V2Check tool
- Place all the VEOs in a single directory
- Copy the V2Check.bat file into the directory
- Edit the V2Check.bat file to change the command line arguments
- Start a MS-DOS command window
- Run the V2Check.bat file in command window

3. Detailed invocation instructions

3.1. Requirements

V2Check requires

- Java 1.8 (later version of Java have not been tested)

3.2. Installation of V2Check

3.2.1. Installing V2Check

Download VERS-V2-Package.zip from the VERS support page at prov.vic.gov.au

Extract the java source code to a suitable place on your hard drive.

3.2.2. Installing Java

V2Check is a Java program and you need to have a Java 1.8 runtime environment or software development kit installed on your computer.

3.2.3. Configuring V2Check

V2Check can be run from the command line in an MS-DOS window, but an alternative is to run it using a MS-DOS bat file (V2Check.bat) included in the Zip file. The default contents of the bat file are shown below:

```
@echo off
set versclasspath="G:\LGVCI\PROV\obiwan\VERS\Vers Coe\Recordkeeping
Standards\Standards Setting\PROS 99-007 Version 2\Code"
java -classpath %versclasspath% V2Check.VEOCheck %*
```

Before running this script, the value of the versclasspath variable must be changed to the directory in which V2Check was installed.

The value of the command line flags can be changed to customise the tests performed.

3.3. Running VEOCheck3

3.3.1. Placement of VEOs

All the VEOs to be tested need to be placed in a single directory.

3.3.2. Test options

The V2Check test tool has a number of options that control the tests that are performed on the VEOs. These options are:

- -*all* Perform all tests. This is equivalent to selecting the options '-extract –signatures –values -virus'.
- -*signatures* Perform the signature tests
- -*values* Perform tests on values
- -*virus* Check for virus infections in the extracted content. This automatically sets the '-extract' option. By default it is assumed that McAfee mcshield service is performing the virus checking. If this is not true, use the '-eicar' option. If viruses are identified, you should first increase the delay ('-d') option to check if the findings are false positives.
- -*eicar* By default it is assumed that the McAfee mcshield service is used to check for viruses. In this case, VEOCheck3 checks that the mcshield service is actually running before and after checking the VEOs. If the McAfee service is not installed, an alternative approach to
- -*extract* Extract the content from the VEOs and place it in files in the current directory. These are labelled by the vers:id attribute associated with the vers:DocumentData element with a file extension from the vers:RenderingKeyword element. A V1 VEO (which doesn't have a vers:id attribute) is labelled as if it had a vers:id attribute and has the prefix 'v1-'. NOTE: this option is really only useful when testing one VEO at a time.
- -*strict* Perform tests strictly according to the standard. This is stricter than the '-da' option.
- -*da* Only test for errors that the digital archive will reject
- -*parseVEO* Parse the original VEO, not a stripped copy of the VEO. The default behaviour of V2Check is to copy the VEO and strip out the document data before parsing it. This makes checking around 10 times faster, but at the cost of making the line numbers reported with parsing errors meaningless. Setting this option will parse the original VEO.
- -*v1.2* Force the test tool to test against version 1 of the standard, irrespective of the value of the version element in the VEO.
- -*v2* Force the test tool to test for version 2 of the standard, irrespective of the value of the version element in the VEO.
- -*verbose* Print more information, in particular, print the values of attributes and values
- -*oneLayer* Only test the outermost layer of values and signatures in a modifiedVEO (version 2) or onion VEO (version 1). This option ignores any errors in the inner layers.

3.3.3. Running V2Check from the command line

V2Check can be run from a MS-DOS (or Unix/Linux) command line. This requires the Java classpath to be configured correctly.

3.3.4. Running V2Check from the V2Check.bat file

Copy the V2Check.bat file from the V2Check directory into the directory that contains the VEOs.

Edit the V2Check.bat file to change the test options.

3.3.5. Run the test script

Using a command window run the V2Check.bat file.

You can test one VEO at a time:

'V2Check test.veo'

or multiple VEOs using the wildcard '*' expansion:

'V2Check *.veo'

The test results are printed to standard out. By default they will appear in the command window. If you a more permanent copy of the output, redirect the standard output to a text file. For example:

'V2Check *.veo > results.txt'

4. Interpreting the output

4.1. Preamble

The standard preamble to the report produced by V2Check has the following appearance:

```
**************************************************************************
*                                                                        *
*                    V E O   T E S T I N G   T O O L                     *
*                                                                        *
*                               Version 1.0                              *
*               Copyright 2005 Public Record Office Victoria             *
*                                                                        *
**************************************************************************

Test run: 2005-11-10 10:30:04+10:00
Testing parameters: All tests (extract, values, signature), Only test outer
layer, Use standard DTD (http://www.prov.vic.gov.au/vers/standard/vers.dtd),
```

The header identifies the VERS testing tool being used and the version. It is followed by the date and time the test is run and the command line options set.

4.2. Test results

The preamble is followed by a set of test results. One set is produced for each VEO tested. The information included in a test result depends on the command line options chosen.

The results for each VEO tested are prefixed by a line of asterixis, and the computer file name of the VEO.

4.3. No errors

Generally, results are only generated for errors (the exception is when the '-values' or '-all' options are selected and '-verbose' output is requested.) Consequently, if no errors are detected for the tests run, the output will look like this:

```
**************************************************************************
Testing '0-DigitalCertificates-17f.veo'
**************************************************************************
```

4.4. XML Errors

Before any other tests are performed, the VEO is always parsed. This test will fail if

- The VEO is not valid XML
- The VEO does not conform to the VERS DTD.

V2Check will not detect if the VEO is not correctly encoded in UTF-8.

Typical parsing errors are:

```
**************************************************************************
Testing '0-DTDExtension-7a.veo'
Test FAILED: SAXException: Fatal Error:
URI=http://www.prov.vic.gov.au/vers/standard/ Line=3:Document root element is
missing.
**************************************************************************
Testing '2-DTDValidation-13a.veo'
```

```
Test FAILED: SAXException: Error:
URI=http://www.prov.vic.gov.au/vers/standard/ Line=193:Element
"vers:RecordMetadata" does not allow "naa:Language" here.
**************************************************************************
```

Errors picked up by the parser are:

- Mis-ordered elements in the VEO (XML document)
- Missing mandatory elements or attributes in the VEO (XML document)
- Additional (non-standard) elements or attributes in the VEO (XML document)

The line numbers reported by V2Check in the parser errors are not useful unless the '-parseVEO' option is used.

When the '-parseVEO' option is selected, V2Check will parse the original VEO. If this option is not selected (the normal behaviour), V2Check will make a copy of the original VEO that does not contain the document data, and then parse the copy. This is 10 to 100 times faster than parsing the original VEO, but means that the line numbers reported in parse errors do not relate to the original VEO. We recommend that you normally run V2Check without the '-parseVEO' option for speed. If a parsing error is reported, rerun V2Check on the failed VEO with the '-parseVEO' option set.

Unfortunately, even with the '-parseVEO' option set, the line numbers may not be accurate as the parser does not always generate accurate line numbers.

4.5. Value Errors

V2Check will check the values contained in certain elements for consistency if the '-values' or '-all' options are selected.

If the '-oneLevel' option is selected, only the outermost (current) layer of a ModifiedVEO or onion VEO is tested. Inner (older) layers are not tested. The default is to test all layers.

If the '-verbose' option is set the values of the elements and attributes will be printed out. This is described in section 4.8.

Note that the Agency Identifier, Series Identifier, and File Identifiers in the VEO cannot be checked for correctness. If these identifiers are incorrect, the VEOs cannot be ingested into the Digital Archive even though V2Check will pass the VEOs.

A typical value error is:

```
**************************************************************************
Testing '2-ElementContent-43ab.veo'
Test FAILED: EMPTY VALUES: The VEO contains the following empty elements:
  <vers:DocumentDate>
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL3: Error in value of element <vers:DocumentDate> (M123). Value is <empty>
      ------------------------------------------------^ Year must match
'yyyy'
```

Note that in this example, the same VEO problem is being reported twice; once as an empty element and then because the date does not match the required format.

4.5.1. Empty Elements

The V2Check will detect elements that have a blank value:

```
**************************************************************************
Testing '2-ElementContent-43aa.veo'
Test FAILED: EMPTY VALUES: The VEO contains the following empty elements:
  <vers:DocumentTitle>
```

A blank value is defined as an

- empty element (e.g. '<vers:DocumentTitle/>')

- element that does not contain any content (e.g. '<vers:DocumentTitle></vers:DocumentTitle>')
- element that only contains white space (spaces, tabs, carriage returns or line feeds) (e.g. '<vers:DocumentTitle> </vers:DocumentTitle>')

4.5.2. VAL1 Errors - Element Errors

Errors marked as 'VAL1' are concerned with element errors. For example:

```
**************************************************************************
Testing '2-VersionAttribute-25b.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL1: Error in element <vers:SignedObject> (M4)
A version 2.0 <vers:SignedObject> (M4) must contain a vers:VEOVersion
attribute
```

In this case, the error is that a vers:SignedObject element in a version 2 VEO must contain a vers:VEOVersion attribute.

The description will contain information about the error that has occurred; in this case that a vers:SignedObject element in a Version 2 VEO does not contain a vers:VEOVersion attribute. The 'M number' is the reference to the element definition in VERS Specification 2.

4.5.3. VAL2 Errors - Attribute Errors

Errors marked as 'VAL2' concern an error with an attribute value. For example:

```
**************************************************************************
Testing '2-VersionAttribute-25a.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL2: Error in attribute vers:VEOVersion (value='1.2') in element
<vers:SignedObject> (M4)
which must be '2.0' to match <vers:Version> (M3) element
```

The first portion of the error message indicates the attribute that caused the error, the value of the attribute, and the element in which the attribute is located. The 'M number' is the reference to the element definition in VERS Specification 2.

This is followed by a detailed description of the problem with the value.

4.5.4. VAL3 Errors - Element Value Errors

Errors marked as 'VAL3' concern an error with an element value. For example:

```
**************************************************************************
Testing '0-VEOType-30j.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL3: Error in value of element <vers:ObjectType> (M6). Value is 'File'
The value of the <vers:ObjectType> (M6) element in a <vers:RevisedVEO> (M158)
element must match the value of the vers:OriginalVEOType attribute in the
<vers:ModifiedVEO> (M156) element (which was Record)
```

The first portion of the error message indicates the element in which the error is located, and the value of the element. Once again the M number is the reference to the element definition in the VERS Specification 2.

This is followed by a detailed description of the problem. In this case it is a problem with consistency between the value of vers:ObjectType and the vers:OriginalVEOType attribute.

4.5.5. VAL4 Errors - Mandatory Element in a Version 2 VEO missing

Errors marked as 'VAL4' are concerned with elements that are mandatory in a Version 2 VEO, but which are not present in the VEO. For example:

```
**************************************************************************
Testing '2-ElementContent-43az.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL4: Element that is mandatory in a version 2 VEO is missing
A version 2.0 VEO must contain at least one <vers:LockSignatureBlock> (M152)
element
```

The VERS DTD covers both version 1 and version 2 VEOs. Consequently, features added in version 2 must be marked as 'optional', even if they are actually mandatory in a version 2 VEO (otherwise version 1 VEOs could not conform to the DTD). The description of the error lists the missing element and includes a reference to the definition of the element in the VERS specification.

The version of the VEO is normally detected automatically using the vers:Version element that is present at the start of every VEO. If necessary, the '-v1.2' or '-v2' options can be used to force V2Check to validate against a particular version.

4.5.6. VAL5 Errors - Mandatory Element in a Version 1 VEO missing

Errors marked as 'VAL5' are concerned with elements that are mandatory in a Version 1 VEO, but which are not present in the VEO. For example:

```
**************************************************************************
Testing '2-V1Encoding-29a.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL270: Document without Long Term Preservation Format
A <Document> (M114) element must contain an <Encoding> (M126) element with a
valid long term preservation format for acceptance into the digital archive.
The Document has no <vers:RenderingKeywords> (M132) elements and so no long
term preservation formats can be identified
VAL5: Element that is mandatory in a version 1 VEO is missing
In a version 1 VEO, a <vers:Document> (M114) element must contain at least one
<vers:Encoding> element
```

The VERS DTD covers both version 1 and version 2 VEOs. Consequently, features added in version 1 must be marked as 'optional', even if they are actually mandatory in a version 1 VEO (otherwise version 2 VEOs could not conform to the DTD). The description of the error lists the missing element and includes a reference to the definition of the element in the VERS specification.

The version of the VEO is normally detected automatically using the vers:Version element that is present at the start of every VEO. If necessary, the '-v1.2' or '-v2' options can be used to force V2Check to validate against a particular version.

4.5.7. VAL6 Errors - Mandatory Element missing

Errors marked as VAL6 mark elements that are optional according to the DTD, but are required by the VERS standard when the VEOs are submitted to PROV. For example:

```
**************************************************************************
Testing '2-XMLDeclaration-3b.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL6: Missing mandatory element
A <vers:VEOIdentifier> (M99) element must contain a <vers:AgencyIdentifier>
(M100) element when submitted to PROV
VAL6: Missing mandatory element
A <vers:VEOIdentifier> (M99) element must contain a <vers:SeriesIdentifier>
(M101) element when submitted to PROV
```

In this case the vers:AgencyIdentifier and vers:SeriesIdentifier are required when the VEOs are submitted to PROV.

The description includes the missing element and includes the reference to the element definition in the VERS specification 2.

4.5.8. VAL7 Errors - Version 2 feature in a Version 1 VEO

Errors marked as VAL7 indicate version 2 features that have been found in a version 1 VEO. For example:

```
**************************************************************************
Testing '2-DTDValidation-48a.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL7: Version 2 feature in a version 1 VEO
A version 1 <vers:SignatureBlock> (M134) element cannot contain a vers:id
attribute
VAL7: Version 2 feature in a version 1 VEO
```

```
A version 1 VEO cannot contain a <vers:LockSignatureBlock> (M152) element
```

The version of the VEO is normally detected automatically using the vers:Version element that is present at the start of every VEO. If necessary, the '-v1.2' or '-v2' options can be used to force V2Check to validate against a particular version.

4.5.9. VAL9 Errors - Missing mandatory attribute in a Version 2 VEO

Errors marked as VAL9 indicate a missing mandatory attribute in a version 2. VEO. For example:

```
**************************************************************************
Testing '1-V2LockSignatureBlock-27b.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL9: Missing mandatory attribute in a version 2 VEO
A <vers:LockSignatureBlock> (M152) element must contain a
vers:signsSignatureBlock attribute
```

The version of the VEO is normally detected automatically using the vers:Version element that is present at the start of every VEO. If necessary, the '-v1.2' or '-v2' options can be used to force V2Check to validate against a particular version.

4.5.10. VAL10 Error – No long term preservation format

Errors marked as VAL10 indicate that one or more documents in the VEO lack a valid long term preservation format (currently a PDF, TIFF, JPEG, or text encoding of the document). For example:

```
**************************************************************************
Testing '2-V1PreservationFormat-21b.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL10: Document without Long Term Preservation Format
A <Document> (M114) element must contain an <Encoding> (M126) element with a
valid long term preservation format for acceptance into the digital archive.
Formats found in this Document are: '.b64; .doc'
```

The PROV digital archive will only accept VEOs which contain a valid long term preservation format for each document in the VEO. That is, each document in the VEO must contain a PDF, TIFF, JPEG, or text encoding (it may contain other encodings as well).

The formats are extracted from the vers:RenderingKeywords (M131) element. If this is not present in the encodings, V2Check will generate a long term preservation format error, even if a valid format exists. This can easily be determined by the error message:

```
**************************************************************************
Testing '2-V1RenderingKeywords-31a.veo'
Test FAILED: INVALID VALUES: The VEO contains the following invalid elements:
VAL10: Document without Long Term Preservation Format
A <Document> (M114) element must contain an <Encoding> (M126) element with a
valid long term preservation format for acceptance into the digital archive.
The Document has no <vers:RenderingKeywords> (M132) elements and so no long
term preservation formats can be identified
VAL6: Missing mandatory element
A <vers:RenderingKeywords> (M132) element must be present in each
<vers:Encoding> (M126) element to allow automated extraction
```

4.6. Signature Errors

V2Check will check the signatures and lock signatures associated with a VEO if the '-signatures' or '-all' options are set. If the '-oneLevel' option is selected, only the signatures on the outermost (current) layer of a ModifiedVEO or onion VEO are tested. The signatures on inner (older) layers are not tested.

4.6.1. SIG8 Errors -- Signature or Lock Signature verification failures

Errors marked as SIG8 indicate that verification of a digital signature or a lock signature failed. A typical error report for a signature or lock signature verification failure is:

```
**************************************************************************
Testing '1-ValidateSignature-42a.veo'
Test FAILED: SIGNATURE: Testing the following signature failed:
```

```
SIG8: Signature verification failed for Signature (vers:id="Revision-1-
Signature-1")
Signature failed verification
  Signature (base64):
eUS7r/o0Y7Ji9jEm79VnJHetONO/Ch/siSSRQx+JGgjgxjHXqvcbP+QnPSBzMfxlbqAxriW6QKidqu
NDLV5tDsWM8QTsC2dcE3338jIoRzo083o019eX5eKMHm0ICH7j1zXV3NoXTjwA0f1zns2u7RJBDmi9
NNdkLuAr3TvG4nQ=<
  Signature (hex):
7944BBAFFA3463B262F63126EFD5672477AD38D3BF0A1FEC892491431F891A08E0C631D7AAF71B
3FE4273D207331FC656EA031AE25BA40A89DAAE3432D5E6D0EC58CF104EC0B675C137DF7F23228
473A34F37A34D7D797E5E28C1E6D08087EE3D735D5DCDA174E3C00D1FD739ECDAEED12410E68BD
34D7642EE02BDD3BC6E274
  Hash of signed object: CD9481794517411B526A069E0290A0C63146DD9C
  Certificate: Subject: CN=Tester, O=PROV, C=AU issued by: O=PROV, C=AU
[
[
  Version: V3
  Subject: CN=Tester, O=PROV, C=AU
  Signature Algorithm: SHA1withRSA, OID = 1.2.840.113549.1.1.5

  Key:  SunJSSE RSA public key:
  public exponent:
    010001
  modulus:
    d26f4ca7 d360ee54 31835f49 5ef0dea5 474f358c d4193811 2247ff08 f401f0ee
    1a9e2fb6 72f467bf 6aafbf1d d38b5ca4 00033846 1eb19349 4273af56 5988ec9d
    76cc71f9 1ce00d7d 10bbfdb3 1f0e3fd4 4e136312 bae9c657 7ea05a2f f96314f6
    ba4b7dbb 422280b4 e73100a1 3b7ba02f 9e28f1ad 815213be 99663896 ead78287
  Validity: [From: Mon Dec 06 12:36:47 EST 2004,
              To: Mon Dec 06 15:23:27 EST 2004]
  Issuer: O=PROV, C=AU
  SerialNumber: [    11]

]
  Algorithm: [SHA1withRSA]
  Signature:
0000: 4A 6D DD A1 CF CD 4E 7E   5C E0 F3 C2 78 A3 51 B4  Jm....N.\...x.Q.
0010: 5E 68 86 8D 98 79 B3 3E   F8 75 D3 6A 13 3A FF C5  ^h...y.>.u.j.:..
0020: 6A 8B 64 F1 77 96 A8 E9   2C 22 23 37 56 57 31 35  j.d.w...,"#7VW15
0030: F8 EE 31 D2 99 E2 9C D7   D2 86 80 F2 C4 E7 53 2A  ..1...........S*
0040: 58 5E 48 C5 8B 96 A3 C5   A0 E4 1F E5 86 CF A1 4E  X^H............N
0050: 43 17 94 C0 98 D7 EC 7B   D0 91 3F E7 D0 0F CE E5  C.........?.....
0060: 8C 99 F7 BA B7 21 FD 27   BA C4 25 50 48 C8 49 3E  .....!.'..%PH.I>
0070: 9A A5 F9 BA BF 06 52 32   37 69 20 F2 14 27 00 64  ......R27i ..'.d

]
```

When a digital signature fails verification, there is no way of determining whether:

- The signed object has been modified
- The signature itself is wrong (e.g. has been calculated incorrectly or been corrupted)
- The public key (from the certificate) is invalid

For this reason, the report of a digital signature includes the following information:

- The vers:id attribute of the failed signature. This can be used to identify which signature failed
- The signature value encoded in Base64. This can be used to confirm which signature failed verification (particularly in a version 1 VEO which does not contain vers:id attributes).
- The signature value in hexadecimal.
- The hash value in hexadecimal. This is the result of applying the nominated hash algorithm to the signed object (this hash value is encrypted using the private key to give the digital signature). If the original hash value, calculated when signing the VEO, is known it can be compared with this value. If the two values are different it suggests that the cause of the verification failure is the use of the wrong hash algorithm, or, more likely, that the wrong portion of the VEO has been signed. If the two values are the same, it

suggests that either the wrong digital signature algorithm was used, or, more likely, the wrong public key (certificate) was used, or the VEO has been corrupted.

- The certificate used to obtain the public key.

4.6.2. SIG9 -- Certificate verification failures

Errors marked as SIG9 indicate that verification of a certificate failed. A typical error report for a certificate verification failure is:

```
**************************************************************************
Testing '1-DigitalCertificates-17d.veo'
Test FAILED: SIGNATURE: Testing the following signature failed:
SIG9: Signature verification failed for Signature (vers:id="Revision-1-
Signature-1")
Certificate 0 failed verification
  Subject of certificate is: CN=Tester, O=PROV, C=AU
  Issuer of certificate is: O=PROV, C=AU
```

Each certificate error contains three basic pieces of information:

- The identity of the certificate that failed (0 is the first certificate in the vers:CertificateBlock)
- The subject of the certificate (i.e. the person or organisation that owns the public key in the certificate)
- This issuer of the certificate (i.e. the organisation that signed the certificate).

If the '-verbose' option is set the contents of the failed certificate is displayed in the error message.

4.7. Encoding Content

If the '-extract' option is set, V2Check will extract and decode the contents of the document data elements.

The contents will be placed into files. The base name of the file will be the contents of the vers:id attribute, with the last file extension taken from the vers:RenderingKeywords element. For example: Revision-1-Document-1-Encoding-1-DocumentData.pdf. Note that if multiple VEOs are tested in one run, the test of a VEO may overwrite the contents of earlier VEOs. Consequently this option is only of use if a single VEO is being tested.

4.8. Full Value and Attribute Dumps

If the '-verbose' option is set with the '-values' or '-all' options, V2Check will produce a dump of the contents of the VEO. This report consists of two lists:

- List of attributes
- List of element values

4.8.1. List of Attributes

The first list contains the attributes that are contained in the VEO. This has been separated from the list of values as this allows the attributes to be clearly identified. The list also serves as a handy summary of the internal structure of the VEO. An example attribute list follows:

```
**************************************************************************
Testing '673-6347-MH_File03-MH_Rec03.xml'
Test SUCCEEDED: ATTRIBUTE VALUES: The VEO contains the following attributes:

 <vers:VERSEncapsulatedObject

ATTRIBUTE:xmlns:vers='http://www.prov.vic.gov.au/gservice/standard/pros99007.h
tm'

ATTRIBUTE:xmlns:naa='http://www.naa.gov.au/recordkeeping/control/rkms/contents
.html'>
  <vers:SignatureBlock
```

```
    ATTRIBUTE:vers:id='Revision-4-Signature-1'>
  <vers:LockSignatureBlock
    ATTRIBUTE:vers:signsSignatureBlock='Revision-4-Signature-1'>
  <vers:SignedObject
    ATTRIBUTE:vers:VEOVersion='2.0'>
   <vers:ObjectContent>
    <vers:ModifiedVEO
      ATTRIBUTE:vers:OriginalVEOType='Record'>
     <vers:RevisedVEO
       ATTRIBUTE:vers:id='Revision-4'>
      <vers:SignedObject
        ATTRIBUTE:vers:VEOVersion='2.0'>
       <vers:ObjectContent>
        <vers:Record>
         <vers:Document
           ATTRIBUTE:vers:id='Revision-4-Document-1'>
          <vers:Encoding
            ATTRIBUTE:vers:id='Revision-4-Document-1-Encoding-1'>
           <vers:DocumentData
             ATTRIBUTE:vers:id='Revision-4-Document-1-Encoding-1-DocumentData'
             ATTRIBUTE:vers:forContentsSeeElement='Revision-1-Document-1-
Encoding-1-DocumentData'>
     <vers:OriginalVEO>
      <vers:SignatureBlock
        ATTRIBUTE:vers:id='Revision-3-Signature-1'>
       <vers:SignedObject
         ATTRIBUTE:vers:VEOVersion='2.0'>
        <vers:ObjectContent>
         <vers:ModifiedVEO
           ATTRIBUTE:vers:OriginalVEOType='Record'>
          <vers:RevisedVEO
            ATTRIBUTE:vers:id='Revision-3'>
           <vers:SignedObject
             ATTRIBUTE:vers:VEOVersion='2.0'>
            <vers:ObjectContent>
             <vers:Record>
              <vers:Document
                ATTRIBUTE:vers:id='Revision-3-Document-1'>
               <vers:Encoding
                 ATTRIBUTE:vers:id='Revision-3-Document-1-Encoding-1'>
                <vers:DocumentData
                  ATTRIBUTE:vers:id='Revision-3-Document-1-Encoding-1-
DocumentData'
                  ATTRIBUTE:vers:forContentsSeeElement='Revision-1-Document-1-
Encoding-1-DocumentData'>
         <vers:OriginalVEO>
          <vers:SignatureBlock
            ATTRIBUTE:vers:id='Revision-2-Signature-1'>
           <vers:SignedObject
             ATTRIBUTE:vers:VEOVersion='2.0'>
            <vers:ObjectContent>
             <vers:ModifiedVEO
               ATTRIBUTE:vers:OriginalVEOType='Record'>
              <vers:RevisedVEO
                ATTRIBUTE:vers:id='Revision-2'>
               <vers:SignedObject
                 ATTRIBUTE:vers:VEOVersion='2.0'>
                <vers:ObjectContent>
                 <vers:Record>
                  <vers:Document
                    ATTRIBUTE:vers:id='Revision-2-Document-1'>
                   <vers:Encoding
                     ATTRIBUTE:vers:id='Revision-2-Document-1-Encoding-1'>
                    <vers:DocumentData
                      ATTRIBUTE:vers:id='Revision-2-Document-1-Encoding-1-
DocumentData'
                      ATTRIBUTE:vers:forContentsSeeElement='Revision-1-
Document-1-Encoding-1-DocumentData'>
             <vers:OriginalVEO>
              <vers:SignatureBlock
                ATTRIBUTE:vers:id='Revision-1-Signature-1'>
```

```
                    <vers:SignedObject
                      ATTRIBUTE:vers:VEOVersion='2.0'>
                     <vers:ObjectContent>
                      <vers:Record>
                       <vers:Document
                         ATTRIBUTE:vers:id='Revision-1-Document-1'>
                        <vers:Encoding
                          ATTRIBUTE:vers:id='Revision-1-Document-1-Encoding-1'>
                         <vers:DocumentData
                           ATTRIBUTE:vers:id='Revision-1-Document-1-Encoding-1-
DocumentData'>
```

4.8.2. List of Elements

The second list displays the contents of the elements in the VEO. If the '-oneLevel' option is selected only the outermost layer of a RevisedVEO or Onion VEO is included in the list.

A partial example of a list of elements is follows. Note that the contents of the following binary elements are suppressed: vers:Certificate, vers:Signature, and vers:DocumentData.

```
Test SUCCEEDED: NORMAL VALUES: The VEO contains the following element values:

 <vers:VERSEncapsulatedObject>
  <vers:VEOFormatDescription>
   <vers:Text> 'Produced according to the Victorian Electronic Records
Strategy, Version 2.0), 31 July 2003. The structure of this record is
represented using Extensible Markup Language (XML) 1.0, W3C, 1998.'
  <vers:Version> '2.0'
  <vers:SignatureBlock>
   <vers:SignatureFormatDescription> 'The contents of this VEO is signed
usingSHA-1 hash algorithm and RSA digital signature algorithm.SHA-1 is defined
in Secure Hash Standard, FIPSPUB 180-1, National Institute of Standards
andTechnology, US Department of Commerce, 17 April1995,
(http://csrc.nist.gov/publications/fips/fips180-1/fip180-1.pdf).The RSA
algorithm (RSASSA-PKCS-v1_5) is definedin PKCS #1 v2.1: RSA Cryptography
Standard, RSALaboratories, 14 June
2002,(ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-1/pkcs-1v2-1.pdf).Details of the
public keys are encoded as X.509certificates in the vers:CertificateBlock
elements.X.509 certificates are define in "Informationtechnology - Open
Systems Interconnection - TheDirectory: Public-key and attribute
certificateframeworks", ITU-T Recommendation X.509 (2000)The signature and
certificates are encoded usingBase64. Base64 is defined in Multipurpose
InternetMail Extensions (MIME) Part One: Format of InternetMessage Bodies,
Section 6.8, Base64 Content-Transfer-Encoding, IETF RFC 2045, N. Freed & N.
Borenstein,November 1996,(http://www.ietf.org/rfc/rfc2045.txt?number=2045)The
signature covers the contents of thevers:SignedObject element starting with
the 'lessthan' symbol of the vers:SignedObject start tag upto and including
the 'greater than' symbol of thevers:SignedObject end tag. Before verifying
thesignature all whitespace (Unicode characters U+0009,U+000A, U+000D, and
U+0020) must be removed from thetext'
   <vers:SignatureAlgorithm>
    <vers:SignatureAlgorithmIdentifier> '1.2.840.113549.1.1.5'
   <vers:SignatureDate> '2005-11-16T01:00:00-10:00'
   <vers:Signer> 'Digital Archive'
   <vers:Signature> (value suppressed)
   <vers:CertificateBlock>
    <vers:Certificate> (value suppressed)
    <vers:Certificate> (value suppressed)
    <vers:Certificate> (value suppressed)
  <vers:LockSignatureBlock>
   <vers:SignatureFormatDescription> 'The contents of this VEO is signed
usingSHA-1 hash algorithm and RSA digital signature algorithm.SHA-1 is defined
in Secure Hash Standard, FIPSPUB 180-1, National Institute of Standards
andTechnology, US Department of Commerce, 17 April1995,
(http://csrc.nist.gov/publications/fips/fips180-1/fip180-1.pdf).The RSA
algorithm (RSASSA-PKCS-v1_5) is definedin PKCS #1 v2.1: RSA Cryptography
Standard, RSALaboratories, 14 June
2002,(ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-1/pkcs-1v2-1.pdf).Details of the
public keys are encoded as X.509certificates in the vers:CertificateBlock
elements.X.509 certificates are define in "Informationtechnology - Open
Systems Interconnection - TheDirectory: Public-key and attribute
```

```
certificateframeworks", ITU-T Recommendation X.509 (2000)The signature and
certificates are encoded usingBase64. Base64 is defined in Multipurpose
InternetMail Extensions (MIME) Part One: Format of InternetMessage Bodies,
Section 6.8, Base64 Content-Transfer-Encoding, IETF RFC 2045, N. Freed & N.
Borenstein,November 1996,(http://www.ietf.org/rfc/rfc2045.txt?number=2045)The
signature covers the contents of thevers:SignedObject element starting with
the 'lessthan' symbol of the vers:SignedObject start tag upto and including
the 'greater than' symbol of thevers:SignedObject end tag. Before verifying
thesignature all whitespace (Unicode characters U+0009,U+000A, U+000D, and
U+0020) must be removed from thetext'
   <vers:SignatureAlgorithm>
    <vers:SignatureAlgorithmIdentifier> '1.2.840.113549.1.1.5'
   <vers:SignatureDate> '2005-11-16T01:00:00-10:00'
   <vers:Signer> 'Digital Archive'
   <vers:Signature> (value suppressed)
   <vers:CertificateBlock>
    <vers:Certificate> (value suppressed)
    <vers:Certificate> (value suppressed)
    <vers:Certificate> (value suppressed)
  <vers:SignedObject>
   <vers:ObjectMetadata>
    <vers:ObjectType> 'Modified VEO'
    <vers:ObjectTypeDescription> 'This object contains a VEO which has been
modified. The modified VEO may be a Record VEO, a File VEO, or another
Modified VEO.'
    <vers:ObjectCreationDate> '2005-07-26T11:04:07+10:00'
   <vers:ObjectContent>
    <vers:ModifiedVEO>
     <vers:DateTimeModified> '2005-11-17T11:07:45Z'
     <vers:RevisedVEO>
      <vers:SignedObject>
       <vers:ObjectMetadata>
        <vers:ObjectType> 'Record'
        <vers:ObjectTypeDescription> 'This object contains a record; that is a
collection of information
that must be preserved for a period of time.'
        <vers:ObjectCreationDate> '2005-07-26T11:04:07+10:00'
       <vers:ObjectContent>
        <vers:Record>
         <vers:RecordMetadata>
          <naa:Agent>
           <naa:AgentType> 'Document Author'
           <naa:Jurisdiction> 'Victoria'
           <naa:CorporateName> 'UR_2M16'
           <naa:PersonalName> 'John Smith'
          <naa:Agent>
           <naa:AgentType> 'Document Author'
           <naa:Jurisdiction> 'Victoria'
           <naa:CorporateName> 'UR_2M16'
           <naa:PersonalName> 'puceblue'
          <naa:RightsManagement>
           <naa:SecurityClassification> 'Unclassified'
           <naa:UsageCondition> 'Copyright State of Victoria 2005'
```