[MS-WCCE]: Windows Client Certificate Enrollment Protocol

This topic lists the Errata found in [MS-WCCE] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS

Atom

Errata below are for Protocol Document Version V43.0 – 2018/09/12.

Errata Published*	**Description**
2020/08/17	In Section 3.2.1.4.3.2.20 PropID = 0x00000014 (CR_PROP_CRLSTATE) "CA CRL State", clarified processing rules to better reflect the behavior that occurs when the client requests the CRL status of all CA signing certificates. Changed from: The CA MUST do the following for each one of the rows in Signing_Cert table: • The CA MUST evaluate the certificate (1) status stored in the Signing_Cert_Certificate column by building its chain based on the specification defined in [RFC3280]. • If the certificate (1) is not valid the CA uses one of the status codes in the following table as the status for this signing certificate. • If the signing certificate is valid, the CA MUST evaluate the base CRL stored in the CRL_Raw_CRL column of the CRL table row where the value of CRL_Name_Id is equal to the row of the preceding signing certificate and verify that it was signed by the key (2) associated with the signing certificate stored in the Signing_Cert_Certificate column. If the signature does not match to the public key of the signing certificate, then the CA MUST return the status 0x01 as specified in the following table. • If the signing certificate is valid and its associated key (2) was used to sign the base CRL stored in the same row, the CA MUST return 0x03 as the status for this signing certificate. Changed to: The CA MUST do the following for each one of the rows in Signing_Cert table: • The CA MUST evaluate the certificate (1) status stored in the Signing_Cert_Certificate column by building its chain based on the specification defined in [RFC3280]. • If the signing certificate is revoked, the CA MUST return the status CA_DISP_REVOKED. • If the certificate (1) index (identified by the Signing_Cert_Certificate column) does not match the key (2) index, the CA MUST return the status CA_DISP_ERROR. • If the certificate (1) index (identified by Signing_Cert_Certificate column) matches the key (2) index, the CA MUST return the status CA_DISP_VALID.
2020/08/17	In Section 3.2.1.4.3.2 ICertRequestD2::GetCAProperty (Opnum 7), revised property index value-descriptions for PropIDs in the last table of this section. Changed from:

PropID	PropIndex MUST be
0x12	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.
0x13	ANY
0x14	ANY
0x1b	ANY
0x1f	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.
0x20	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.
0x25	ANY
0x26	ANY
0x27	ANY
0x2B	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.

Changed to:

PropID	PropIndex MUST be
0x12	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.
0x13	ANYThe minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.
0x14	ANYThe minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.
0x1b	ANYThe minimum index is 0. The maximum index is one less than value of the Config_CA_KRA_Cert_Count datum.
0x1f	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.
0x20	The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.
0x25	ANYThe index corresponds to a particular CA signing certificate. Since the last CA signing certificate cannot have a forward cross certificate, the minimum index is 0 and the maximum index is two less than the count of rows in the Signing_Cert table.
0x26	ANYThe index corresponds to a particular CA signing certificate. Since the first CA signing certificate cannot have a backward cross certificate, the minimum index is 1 and the maximum index is one less than the count of rows in the Signing_Cert table.

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td><table><tr><td>0x27</td><td>ANYThe minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table.</td></tr><tr><td>0x2B</td><td>The minimum index is 0. The maximum index is one less than the count of rows in the Signing_Cert table. An index of 0xFFFFFFFF is allowed and indicates the maximum valid index.</td></tr></table></td></tr>
<tr><td>2020/08/17</td><td>In Section 3.2.1.4.3.2.20,clarified the client request for all CA signing certificates. Also corrected the processing instruction to specify the return value from the CA as 0x03 when the signing certificate is valid and its associated key was used to sign the base CRL stored in the same row.
Changed from:
"The client has requested to identify which signing certificate is associated with the key (2) used to publish CRLs.
The CA MUST do the following for each one of the rows in Signing_Cert table:"
. . .
● "If the signing certificate is valid and its associated key was used to sign the base CRL stored in the same row, the CA MUST return0x00 as the status for this signing certificate."
Changed to:
"The client has requested the CA signing certificate status for all CRLs."
The CA MUST do the following for each one of the rows in Signing_Cert table:"
. . .
● "If the signing certificate is valid and its associated key was used to sign the base CRL stored in the same row, the CA MUST return0x03 as the status for this signing certificate."</td></tr>
<tr><td>2020/08/03</td><td>In Section 3.2.1.4.3.2, ICertRequestD2::GetCAProperty (Opnum 7), added missing PropID entry value (0x2D) at the end of last Table in this section, which defines values that MUST be set for PropIndex and PropType parameters for each property value passed via the PropID parameter.
Changed from:
<table><tr><td>0x2C</td><td>0x00000000</td><td>0x00000004</td></tr></table>
Changed to:
<table><tr><td>0x2C</td><td>0x00000000</td><td>0x00000004</td></tr><tr><td>0x2D</td><td>0x00000000</td><td>0x00000004</td></tr></table></td></tr>
<tr><td>2020/08/03</td><td>In Section 3.2.1.4.3.2.29, PropID = 0x0000001D (CR_PROP_TEMPLATES) "Configured Certificate Templates", revised the processing instructions to specify an updated server return value for the PropID = 0x0000001D property in the GetCAProperty method, consisting of a string containing pairs of the template name and OID separated by new lines.
Changed from:</td></tr>
</table>

<table>
<tr><th>Errata Published*</th><th>Description</th></tr>
<tr><td></td><td>
<p>The client requested to know the list of certificate templates that are configured for this CA.</p>
<p>The server MUST return an empty CERTTRANSBLOB (section 2.2.2.2) structure.</p>
<p>Changed to:</p>
<p>The client requested to know the list of certificate templates that are configured for this CA.</p>
<p>The server MUST return a string containing the list of templates supported by this CA, with one pair of name and string OID for each template and separated by new lines, as in the format that follows:</p>
<p>"name1\nOID1\nname2\OID2...\nnameN\nOIDN\n\0"If the template does not have an associated OID (Win2k domain), there will be an empty string in its place.</p>
</td></tr>
<tr><td>2019/12/16</td><td>
<p>In Section 3.1.2.4.2.2.2.8 , Certificate.Template.msPKI-Private-Key-Flag, added missing 'CT_FLAG_HELLO_LOGON_KEY' flag and description to the processing rules table. Also added new informative reference [MSDOCS-WHfB] to the description for the missing flag</p>
<p>Changed from:</p>
<table>
<tr><td>0x000001000
CT_FLAG_ATTEST_PREFERRED
*</td><td>This flag instructs the client to generate a certificate request as explained in section 3.1.1.4.3.4.1.1 if the Client_HardwareKeyInfo and Client_KeyAttestationStatement ADM elements are not empty (as described in section 3.1.2.4.2.2.2.2).</td></tr>
</table>
<p>Changed to:</p>
<table>
<tr><td>0x000001000
CT_FLAG_ATTEST_PREFERRED
*</td><td>This flag instructs the client to generate a certificate request as explained in section 3.1.1.4.3.4.1.1 if the Client_HardwareKeyInfo and Client_KeyAttestationStatement ADM elements are not empty (as described in section 3.1.2.4.2.2.2.2).</td></tr>
<tr><td>0x00200000
CT_FLAG_HELLO_LOGON_KEY
*</td><td>This flag instructs the client to generate a certificate request for the Windows Hello Logon key. For more information about Windows Hello for Business, see [MSDOCS-WHfB].</td></tr>
</table>
</td></tr>
</table>